LATEST VERSION
DON MANKIN

TABLE OF CONTENTS

SECTION-0 - FOREWARD 1

i.0.0 MANUAL FORMAT 1

i.1.0 OPERATING SYSTEMS SUPPORTED 2
i.2.0 COMPATIBILITY WITH OTHER DRIVES 2
i.2.1 PERCOM DOUBLE SIDED DRIVES 2
i.2.2 MICRO MAINFRAME DOUBLE SIDED DRIVES 2
i.3.0 POLICIES AND PROCEDURES 3

SECTION-1 - OPERATIONS MANUAL 4

1.1.0 BRINGING UP THE SYSTEM FOR THE FIRST TIME . . 4
1.2.0 BOOTING THE OPERATING SYSTEM DISKETTE 4
1.3.0 SELECTION OF THE LINE PRINTER OPTION 5
1.4.0 MAKING A DOUBLE DENSITY DISKETTE 6
1.4.1 WITH A SINGLE DRIVE SYSTEM 6
1.4.2 WITH A DUAL DRIVE SYSTEM 7
1.5.0 FIGURES AND ILLUSTRATIONS 7
1.6.0 WHEN THINGS GO WRONG 8
1.7.0 SW1 AND SW2 SAMPLE CONFIGURATIONS 8

SECTION 2 - TECHNICAL MANUAL 10

2.1.0 PORT DESCRIPTION OF THE PCP-80 AND FPC-80 . . 10
2.1.1 HARD DISK HOST ADAPTER PORT USAGE 10
2.1.2 DECRIPTION OF PCP-80 AND FPC-80 PORTS . . . 11
2.2.0 ROM SWITCH SW1 DEFINITIONS - NON-ATARI MODE . 13
2.3.0 ROM SWITCH SW1 & SW2 DEFINITIONS - ATARI MODE 14
2.3.1 SWITCH SW1 CONFIGURATION - ATARI MODE . . . 15
2.3.2 SWITCH SW2 CONFIGURATION - ATARI MODE . . . 16
2.3.3 PROGRAMMING NOTES 16
2.4.0 XY DRIVE CONFIGURATION BIT MAP 17

SECTION 3 - PROGRAMMING MANUAL 18

3.1.0 UTILITY COMMANDS SUPPORTED 18
3.1.1 SOFTWARE COMMANDS SUPPORTED VIA THE LINE PRINTER
3.1.2 COMMAND FORMAT 18
3.1.3 OP-CODES SUPPORTED 19
3.1.4 OP CODE DESCRIPTIONS 19
3.2.0 PROGRAMMING EXAMPLES 20
3.3.0 LINE PRINTER DEVICE DESCRIPTION BLOCK 21
3.4.0 HARD DISK CONTROL BLOCK 21
3.4.1 HARD DISK DEVICE BLOCK 21
3.4.2 BIT MAP OF RELATIVE BYTE 02 22
3.4.3 BIT MAP OF RELATIVE BYTE 05 22
3.4.4 OP CODE DEFINITIONS 23
3.5.0 LOGICAL ADDRESS COMPUTATION 26
3.5.1 FLOPPY DISK LOGICAL ADDRESS COMPUTATION . . 26

3.5.2 HARD DISK LOGICAL ADDRESS COMPUTATION . . . 26

APPENDIX A 27

A.1.0 APPLICATION NOTE - 1771/1797 27

APPENDIX B 28

B.1.0 APPLICATION NOTE - BAUD RATE SELECT 28
B.1.1 19.2 K BAUD CLOCK WAVEFORM 28
B.1.2 38.4 K BAUD CLOCK WAVEFORM 29

APPENDIX C 30

C.1.0 UTILITY PROGRAMS SUPPLIED BY MICRO MAINFRAME 30
C.1.1 CONVERT.V1 30
C.1.2 FASTCOPY.V11 32
C.1.3 PRTSEC.V11 - PRINT SECTORS 33
C.1.4 PRTRK.V11 - PRINT TRACKS 33
C.1.5 SETDAM.V11 - CHANGE DAM BYTE 33
C.1.6 DISASSEM.V1 34
C.1.7 INHANC.V11 34
C.1.8 STFORMAT.V11 35
C.1.7 SPEED.V11 36

MICROZAP (TM) INSTRUCTIONS M1

MF-XX8X SERIES

DISK DRIVE FOR ATARI

BY MICRO MAINFRAME

SECTION-0 - FOREWARD REVISION 0.01

i.0.0 MANUAL FORMAT

Congratulations!! You have achieved a new dimension in computing power by the addition of a Micro Mainframe disk drive to your Atari computer system. We think that you will find the difference in performance between the Micro Mainframe disk drive and other disk drives for the Atari is like comparing the performance of a Boeing 707 to that of an F-15 Fighter.

This manual is valid for all Micro Mainframe disk drive Models MF-XX8X. The nomenclature for the MMF Atari Disk drive is as follows:

where:

XX = 16 for a 16K unit; 64 for a 64K unit.
8 designates that the processor is a Z80.
X is 1 for a single drive unit and 2 for a dual drive unit.

Therefore, an MF-1681 designates a 16K single drive unit; an MF-6481 designates a 64K single drive unit; an MF-6482 designates a 64K dual drive unit, etc.

The manual has been divided into sections to make it more understandable. The OPERATIONS MANUAL (Section-1) will provide the novice user with the basic information necessary to operate his new disk drive with a minimum expenditure of time and effort. YOU MUST READ SECTIONS 0 AND 1 OF THIS MANUAL BEFORE YOU ATTEMPT TO OPERATE YOUR DISK DRIVE.

Most of you may never desire to know any more about your disk drive than the information provided in the OPERATIONS MANUAL, and for those individuals, the information provided therein may comprise the total information required. Some of you, however, may wish to know the innermost workings of your system. For those users we have provided Section-2 and Section-3. It is our hope that these sections will give you the desired insight into the system.

Section-1 covers six important areas; (1) Using the drive as shipped; (2) Booting the operating system diskette; (3) Configuration of the line printer switch; (4) Making a Double Density diskette; (5) Diagrams and Illustrations; and (6) When Things Go Wrong.

Section-2 is a discussion of the PORT DESCRIPTIONS AND DEFINITIONS for the PCP-80 computer and the configuration procedures for DIP switches SW1 & SW2.

SECTION-3 is the PROGRAMMING MANUAL, which will explain how to unleash some of the hidden powers of your disk drive. In order to fully understand this part of the manual, you must obtain copies of each of the following manuals: Z-80 Technical Manual, Z-80 Assembly Language Programming Manual, and and the Western Digital applications notes for the WD179X and WD1771 disk controllers. These manuals are available from the respective manufacturers. SECTION-3 describes the command structure which may be sent to the PCP-80 via the line printer port.

APPENDIX A considers certain incompatibilities between the WD1771 disk controller (used in the Atari 810 (TM) disk drive) and the WD1797 as used in the Micro Mainframe disk drive.

 ©1983, Micro Mainframe

APPENDIX B explains the Double Baud Rate timing, while APPENDIX C explains the programs contained on the Micro Mainframe utility diskette shipped with every drive.

YOU SHOULD FIRST CONNECT THE DISK DRIVE TO YOUR ATARI WITHOUT ANY CHANGES TO THE CONFIGURATION TO DETERMINE IF IT IS WORKING PROPERLY IN THE 810 EMULATOR MODE.

COPYRIGHT NOTICE

This manual contains several references to the proprietary or trademarked products of other manufacturers. These include Atari, Percom Data, Optimized Systems Software, and Wordmark Systems. This paragraph serves as the required notification and acknowledges the products of the other vendors mentioned herein.

i.1.0 OPERATING SYSTEMS SUPPORTED

The Micro Mainframe MF-XX8X series disk drives currently support three operating systems. These are: ATARI DOS 2.0S, OS A+ by OPTIMIZED SYSTEMS SOFTWARE (VERSIONS 4.0 & 4.1), and MYDOS by WORDMARK SYSTEMS. YOU MUST PURCHASE ONE OF THESE OPERATING SYSTEMS FROM YOUR DEALER IN ORDER TO MAKE YOUR MF-XX8X DISK DRIVE OPERATIONAL!

i.2.0 COMPATIBILITY WITH OTHER DRIVES

The Micro Mainframe MF-XX8X series of disk drives is fully compatible with the Atari 810 disk drive and will also emulate the special commands used with the Percom disk drive. Any software that uses the Percom special commands will function properly on the Micro Mainframe disk drive. The Micro Mainframe disk drive is completely compatible with the Percom unit with two exceptions as discussed below.

i.2.1 PERCOM DOUBLE SIDED DRIVES

Percom elected to use a different convention for their double sided feature than the one used by Micro Mainframe. Percom records sectors 1-720 on side A of the diskette starting at the outside edge of the diskette and working toward the inside. Then they switch to the top head and begin recording sectors 721-1441 on side B of the diskette beginning at the inside and working to the outside edge of the diskette. This allows for only 18 consecutive sectors available per head movement. In order to read sector 19, Percom must step the head back to track 1, which requires 12 milliseconds for the physical movement of the head, and additionally up to 200 milliseconds once the head has been stepped to retrieve the data.

i.2.2 MICRO MAINFRAME DOUBLE SIDED DRIVES

The Micro Mainframe double sided configuration assigns sectors 1-18 to Track 0, Side A of the diskette and sectors 19-36 to Track 0, Side B of the diskette for a total of 36 sectors on each track. Sector 37 would therefore begin on Track 1 of Side A. This means that we can move the disk head once and have access to 36 sectors per head movement instead of only 18 per head movement as Percom does. We feel that this method is considerably more efficient since it reduces head movement by presenting twice as many sectors at one time without requiring head movement. This yields faster data access on consecutive sector reads and writes.

One other area of incompatibility may exist although we have not been able to confirm this. Wordmark Systems has reserved one bit to select the Percom implementation for 8" disk operation. Micro Mainframe is unaware of any intent by Percom to support 8" disk operation, and has elected to continue using our method of selecting 8" disk operation.

 ©1983, Micro Mainframe

i.3.0 POLICIES AND PROCEDURES

PLEASE NOTE: THE WARRANTY CARD SUPPLIED WITH YOUR MF-XX8X SERIES DISK DRIVE MAY INCORRECTLY STATE THAT THE WARRANTY PERIOD IS 180 DAYS. THIS IS DUE TO PRINTING ERROR. THE CORRECT WARRANTY PERIOD IS 90 DAYS AS WE HAVE PREVIOUSLY PUBLISHED IN OUR LITERATURE. WE HAVE MARKED OUT THE INCORRECT PERIOD WITH RED INK, AND WE HAVE INSERTED THE CORRECT ONE BY HAND. WE APOLOGIZE FOR ANY CONFUSION THAT THIS MAY CREATE.

Micro Mainframe's is open for TELEPHONE ORDERS ONLY from 8:30 AM to 5:00 PM Pacific Time. Calls will be accepted only during those hours. MMF DOES NOT ACCEPT COLLECT PHONE CALLS FOR ANY REASON!! WE HAVE NO FACILITY FOR HANDLING WALK-IN CUSTOMERS.

MMF will not accept products returned for any reason without an RMA number affixed to the front of the package returned. YOU must obtain this number from our Customer Service Department. All incoming shipments must be FOB Point of Origin. MMF does not accept COD shipments. No refunds for any shipping charges to MMF will be made under any circumstances.

PLEASE NOTE: Information in this publication is based upon specifications and data believed correct at the time of publication. Micro Mainframe reserves all rights to make any changes in specifications and models without prior notice to any party as design improvements are introduced. Micro Mainframe reserves the right to implement price changes at any time without prior notice to any party.

MICRO MAINFRAME
11325-A Sunrise Gold Circle
Rancho Cordova, CA 95670
(916) 635-3997

 ©1983, Micro Mainframe

SECTION-1 - OPERATIONS MANUAL

1.1.0 BRINGING UP THE SYSTEM FOR THE FIRST TIME

1.1.1 After passing an exhaustive checkout procedure, your MICRO MAINFRAME (MMF) disk drive has been factory programmed to emulate a standard Atari 810 prior to shipping: ie - Single Density, Drive D1:, No Line Printer operation, and the Automatic Density Recognition mode is ON.

1.1.2 Carefully unpack your MMF disk drive from the shipping carton. SAVE THE CARTON, AS YOU MUST USE IT TO RETURN THE DISK DRIVE TO MMF IN THE UNLIKELY EVENT THAT YOUR UNIT NEEDS REPAIRS. MMF WILL ACCEPT DRIVES FOR REPAIR ONLY IF THEY ARE RETURNED IN THE ORIGINAL SHIPPING CARTON.

The following procedure assumes that you have a Micro Mainframe disk drive attached to your Atari computer as the only disk drive. If you will be using the MMF drive in conjunction with drives provided by other vendors, we still recommend that you initially connect the MMF drive as if it were the only drive on the system to check out the drive operation with your particular computer. Once you have determined that the MMF drive is functioning properly with your system, you may then skip to Sections 2 & 3 for some sample configuration procedures using the advanced features of the Micro Mainframe MF-XX8X series disk drives.

1.1.3 Referring to FIGURE 1.5.1, locate the cable marked TO ATARI COMPUTER on your MF-XX8X disk drive and connect it to the expansion socket on the right side of your 400 or 800 computer or on the rear panel of your 1200. This is the only connection to your Atari computer system required by the MMF disk drive.

a. Connect the AC POWER CABLE to a suitable source of AC power (115 VAC).

b. Turn on the Drive Power Switch (Designated SW4).

1.2.0 BOOTING THE OPERATING SYSTEM DISKETTE

1.2.1 You may boot up with or without the Basic Cartridge installed in your Atari computer (Left socket on the 800 computer system).

1.2.2 NOTE: MMF IS CURRENTLY SUPPLYING DISK DRIVES BY TWO DIFFERENT MANUFACTURERS. They differ only in the method used to open and close the door. One type has a rotating handle on the front of the drive and the other has a spring loaded push bar to close and open the door. Both drive types are identical in performance, and either may be used interchangeably by MMF.

Open the door on the MMF disk drive by turning the handle counter-clockwise to the HORIZONTAL position or by depressing the push bar depending upon your drive type. Insert your operating system diskette (not supplied) into the disk drive referring to FIGURE 1.5.2. If you have not obtained one, they may be purchased either from Atari, OSS or WORDMARK SYSTEMS.

After you have inserted the diskette, close the door on the disk drive either by turning the handle clockwise to the VERTICAL position or by pulling out on the handle with your fingers as you close the door.

1.2.3 You are now ready to tell the disk drive to BOOT the operating system diskette. BOOT literally means BOOTstrap, and as used here means to load a special program from the master diskette (the one you purchased from Atari, OSS or Wordmark Systems). The boot software initializes certain parameters in your Atari computer, and then loads the operating system into the memory of your Atari. We tell the

Atari to BOOT the disk drive by performing one of the two operations described below. YOU MUST FIRST CONNECT THE DISK DRIVE TO THE ATARI AND TURN ON THE DISK DRIVE POWER SWITCH BEFORE YOU ATTEMPT TO BOOT THE DOS.

a. Open and then close the cartridge slot on the Atari 400 or 800 as if to install a cartridge (not applicable to the 1200).

b. Turn the Atari power switch off and then back on (This switch is located on the right side of the Atari 400 or 800 computer system and at the left rear for the 1200).

Each time that you open and close the cartridge door or turn the Atari power switch off and back on, the disk drive will attempt to BOOT the diskette you have inserted into the disk drive, PROVIDED THAT THE DISK DRIVE IS CONNECTED AND THE POWER IS TURNED ON.

c. Insert a copy of your Master DOS diskette at this time (don't forget to close the door on the disk drive), turn up the volume on your TV set and give the command to BOOT by one of the two methods outlined above. You should hear a beeping sound emanating from the TV speaker indicating that the Atari is loading the operating system from your disk drive. Upon completion of the load you will get one of two messages on the screen.

1. If your Basic Cartridge IS installed, the computer will load the Operating System and return to Basic and display READY. You should type DOS and hit the <RETURN> key to get into the operating system. You will be given a Menu of operations that you can perform at this time. Please refer to the Atari, OSS or Wordmark Systems DOS manual for further information on the use and features of the operating system.

2. If the Basic Cartridge IS NOT installed, the computer will go immediately to DOS and present the menu described in 1 above. You should now become familiar with the procedures outlined in the manual included with your operating system.

d. Your disk system should now be operational. If the system did not perform as indicated, you have probably failed to perform one of the steps in the correct order. Recheck your work before going into panic. If you still have problems then please refer to Paragraph 1.6.0 of the Operations Manual entitled WHEN THINGS GO WRONG!

If you are not familiar with disk operating procedures, we refer you to the DOS manual that came with your operating system. We recommend that you make several back up copies of your operating system at this time and then put your master diskette in a safe place. As this manual is not intended to replace your DOS manual, you should now become familiar with all of the procedures for your particular operating system. Once you understand how the DOS operates, you may wish to explore the other sections of this manual.

1.3.0 SELECTION OF THE LINE PRINTER OPTION

1.3.1 Configuring your MMF disk drive to support the Centronics compatible line printer is as simple as the flip of a switch. Refer to FIGURE 1.5.1 for the location of the configuration switches. Examine Switch SW1 and locate switch 6 (the sixth switch from the left). As shipped from the factory, this switch is turned off. To enable the line printer option, you must turn this switch on. The positions of this switch are shown in the SAMPLE CONFIGURATION CHARTS beginning at PARAGRAPH

 ©1983, Micro Mainframe

1.7.0. Don't forget to depress the <RESET> SWITCH SW3 after you have changed the settings of any of these switches.

1.4.0 MAKING A DOUBLE DENSITY DISKETTE

Contrary to what you may have been told, Atari 2.0S DOS will run in the Double Density mode. It is very simple to accomplish if you follow the instructions given below. PLEASE NOTE: Atari formats diskettes with 128 byte sectors in single density and 256 byte sectors in double density. When you do a directory on a newly formatted double density diskette, the ATARI DOS will report the same number of sectors available, thereby giving the misconception that the procedure did not work properly. When you CONVERT your single density programs over to double density, you will find that they now occupy approximately one half the number of sectors as they had previously occupied under single density.

1.4.1 WITH A SINGLE DRIVE SYSTEM

Single drive users must use the program SETDBL located on the program disk supplied by MMF. This is the preferable method for making double density diskettes as it requires no changes to your drive default settings. Owners of dual drive systems may use either this procedure or the one noted in Paragraph 1.4.2 below. To make a double density System disk, please follow the procedure below:

1. Refer to Paragraph 1.3.0 and configure SW1-6 to use the line printer. YOU MUST ENABLE (TURN ON) THIS SWITCH EVEN IF YOU DO NOT HAVE A LINE PRINTER OR YOU WILL FORMAT A SINGLE DENSITY DISKETTE INSTEAD OF A DOUBLE DENSITY DISKETTE.
2. DEPRESS SW3 ON THE BACK OF THE MMF DISK DRIVE TO READ THE DIP SWITCH SETTINGS.
3. Remove your BASIC cartridge at this time, insert your diskete containing Single Density Atari DOS 2.0S, and boot your Atari.
4. When the Menu appears, remove the Atari diskette and insert the Micro Mainframe utility diskette you received with your disk drive. **Please Note: this diskette does not contain an operating system and will not boot by itself.**
5. Type L and when it asks for the program name, type D1:SETDBL.V1 This will load the SETDBL.V1 program from the Micro Mainframe Utility diskette.
6. The program will instruct you to insert the blank (unformatted) diskette that you wish to format to double density into drive D1:, leaving the door of the disk drive open (you will be prompted in case you forget). The drive will select and configure itself for double density. If you forgot to enable the line printer switch SW1-6 or if the drive did not properly configure, you will get a message advising you to configure the switch before proceeding.
7. SETDBL.V1 will proceed to format your diskette to double density and regenerate the operating system on the newly formatted double density diskette. In order to boot up your new double density diskette, you must turn the Atari off and back on in order to boot the system. You may now load up the CONVERT.V11 copy program supplied on the utility diskette convert your non-protected files over to double density. You will find instructions regarding the use of this program in APPENDIX C. NOTE: **this program will not work for copy protected diskettes or for DOS.SYS or DUP.SYS files.** Diskettes formatted by SETDBL.V11 will say DOS 2.0M instead of DOS 2.0S.

 ©1983, Micro Mainframe

1.4.2 WITH A DUAL DRIVE SYSTEM

You must remove your Basic cartridge from the computer prior to performing the following procedure. This procedure is for use with a double drive MMF unit only.

1. Lift SW2-3 up to set D2: to double density. This switch is located on the rear panel of the drive **(See FIGURE 1.5.1 for the location of the configuration and power switches).**
2. Depress the <RESET> switch (SW3) at the rear of the MMF disk drive to read the new configuration **(see FIGURE 5.1.1 for the location of this switch). D1: will be set for single density operation and D2: will be set for double density operation.**
3. **Insert a copy of your master DOS diskette in drive D1 and close the door by turning the handle to the down position. Leave the door open on drive D2. Boot the DOS diskette by turning your Atari off and back on to cause the Atari to configure drive D1 single density and drive D2 double density.**
4. **Insert a blank (unformatted)** diskette in drive D2 and CLOSE the door.
5. Issue the I command from Atari DOS specifying D2: when asked which drive to format. The Atari will now format the diskette in D2: to double density.
6. At the DOS ready prompt, issue the H command. Specify D2: when it asks for the drive. The Atari will copy the system files onto the double density diskette you just formatted in drive D2. That is all there is to it. You may desire to format several double density diskettes at this time and transfer your single density files over to the doubl density diskettes. PLEASE NOTE: you will NOT be able to copy any of the COPY PROTECTED OR SELF BOOTING diskettes onto double density with this program. You are on your own if you try to move such programs over to double density, as a discussion of this topic is beyond the scope of this manual.

1.5.0 FIGURES AND ILLUSTRATIONS

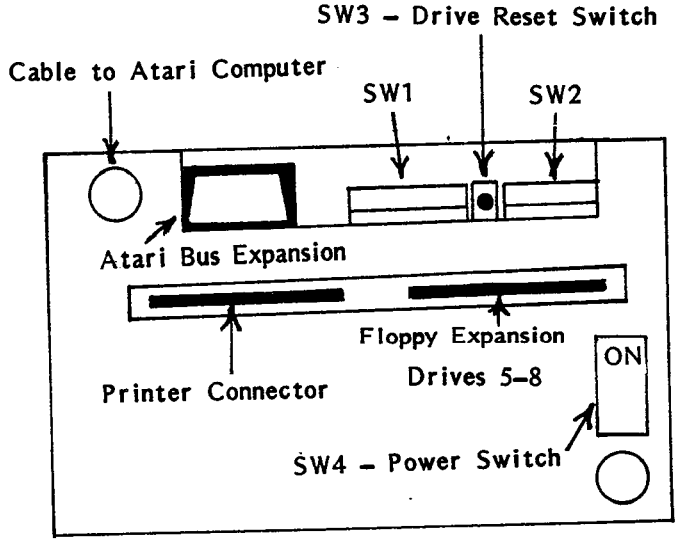


FIGURE 1.5.1

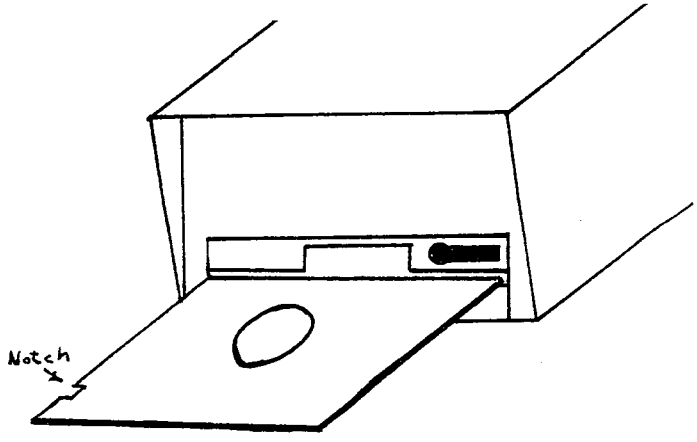


FIGURE 1.5.2

 ©1983, Micro Mainframe

1.6.0 WHEN THINGS GO WRONG

If you have read this far, then you are, most likely, having problems with one or more of the procedures outlined in the manual above. If this is the case DO NOT PANIC as you have probably overlooked a small, but important step in one of the procedures outlined previously. This section of the manual is designed to help you locate your problem.

Since even the most basic computer system is infinitely complex when compared to other electronic systems, we must rationally approach the problem. If we do not, we may end up in a circular loop and never locate the real problem.

When you have a system level problem, it is extremely important to define what is operating properly and what is not. This is the key to the successful troubleshooting of a complex system.

Do not overlook such simple factors as the following: Check to see if the door on the disk drive or drives is closed; Check all cables to see if they are properly connected; Check the diskette to make certain you have properly inserted it into the disk drive; Check the configuration switches to make certain that you have configured the system for what you want to do; Check to see if the power is on to both the Atari and the MF-XX8X disk drive. Press the <RESET> switch (SW3) located at the back of the disk drive to cause the PCP-80 CPU to re-read the configuration switches. Insert a diskette containing the DOS files in the appropriate density for your configuration into D1 and close the door. Turn on the power to the Atari and the diskette should boot. If it does not, then check the Serial Cable that goes from the disk drive to the Atari to ascertain if it is properly seated in the Atari's expansion connector.

Most likely, you have forgotten to perform one of the required steps in the proper order, and you will probably fix the problem during your checkout.

Please remember that on a dual drive system, if one of the drives is working in any mode, it is a good indication that the CPU and Controller in the MF-XX8X are working properly. If the second and successive drives are not working, the problem is most likely the drive mechanism itself, the cable connections, the power supply for the additional drive (if you are using a power supply external to the one in your MF-XX8X unit), or with your configuration procedure.

Before you call the Micro Mainframe Customer Service Department, please perform the procedures outlined above. The technician will ask you if you have performed the tests noted above, and your answers could be important to the resolution of your problem. Please remember that the power supply in the MF-XX8X will power only two disk drives. If you attempt to run more than two disk drives and damage the power supply, the damage incurred will not be covered under the Micro Mainframe Limited Warranty.

If you have attempted the procedures outlined above and still cannot make your disk drive function, please call the Micro Mainframe Customer Service Department at the number listed below. Please have the serial number of your, drive, your invoice number and the date of purchase available for the technician when he answers.

1.7.0 SW1 AND SW2 SAMPLE CONFIGURATIONS

Some individuals report having difficulty understanding the configuration procedure for the drive configuration switches SW1 and SW2. The intent of this paragraph is to amplify the material presented in Section-2. You will soon see how easy it is to make your drive functional in only a few of the many possible configurations of which it is capable. Please note that a switch is considered ON when it is in the UP position. This condition is designated in the charts below by a 1. A switch is considered OFF when it is in the DOWN position. This condition is designated in the charts below by a 0. Although the dip switches are marked to indicate that UP is OPEN, the PCP-80 treats the switches as closed, since we use negative true logic when we read the condition

of these switches. ANY TIME YOU CHANGE THE CONDITION OF ANY OF THESE SWITCHES, YOU MUST DEPRESS RESET SWITCH SW3 BEFORE THE DRIVE WILL RECONFIGURE ITSELF TO THE NEW SETTINGS.

The following are Single Density, 5 1/4" configurations with one MMF Master drive and up to three slave drives. Slave drives are those that are attached to the INTERNAL disk controller bus unless otherwise specified. If two MMF master units are connected to a single computer, then each is treated as a separate unit and not as a slave.

CONFIGURATION	SW1	SW2
SWITCH NUMBER	1 2 3 4 5 6 7 8	1 2 3 4 5 6 7 8
One Master, No Slaves, No LPT	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0
One Master, No Slaves, LPT	0 0 0 0 0 1 0 0	0 0 0 0 0 0 0 0
One Master, one slave, LPT	0 0 0 1 0 1 0 0	0 0 0 0 0 0 0 0
One Master, Two Slaves, LPT	0 0 0 0 1 1 0 0	0 0 0 0 0 0 0 0
One Master, Three Slaves, No LPT	0 0 0 1 1 0 0 0	0 0 0 0 0 0 0 0

The following configurations are Double Density configurations. They are otherwise identical to the configurations enumerated above.

CONFIGURATION	SW1	SW2
SWITCH NUMBER	1 2 3 4 5 6 7 8	1 2 3 4 5 6 7 8
One Master, No Slaves, No LPT	0 0 0 0 0 0 0 0	1 0 0 0 0 0 0 0
One Master, No Slaves, LPT	0 0 0 0 0 1 0 0	1 0 0 0 0 0 0 0
One Master, One Slave, LPT	0 0 0 1 0 1 0 0	1 0 0 0 0 0 0 0
One Master, Two Slaves, LPT	0 0 0 0 1 1 0 0	1 0 0 0 0 0 0 0
One Master, Three Slaves, No LPT	0 0 0 1 1 0 0 0	1 0 0 0 0 0 0 0

The following configuration is useful if you have an ATARI or other drive that you wish to use in conjunction with an MMF disk drive. Two cases are given of which the first is valid when the non-MMF drive is designated as drive D1: and the MMF drive is designated as D2:. The second is with the MMF drive at D1:, and the non-MMF drive is at D2:. The MMF drive is shown programmed for double density operation in the first two of these examples. If you wish to operate two MMF Master drive units, you would use both of these configurations, assigning each MMF drive as a separate unit.

In the third example, we have placed a two drive MMF system at drive D5:, the first of which is programmed for 5 1/4" double density operation, and the second of which is programmed for 8" single density operation. This allows us to use drive D5: and D6: on the system. In order to use this configuration you will have to tell your DOS that you have six drives on the system and define those drives using the DEFINE DEVICE command as outlined in Section-3 of this manual. You will also need an operating system that allows you to use 8" diskettes.

CONFIGURATION	SW1	SW2
SWITCH NUMBER	1 2 3 4 5 6 7 8	1 2 3 4 5 6 7 8
MMF Drive @ D2:	1 0 0 0 0 0 0 0	1 0 0 0 0 0 0 0
MMF Drive @ D1:	0 0 0 0 0 0 0 0	1 0 0 0 0 0 0 0
MMF Drives @ D5: D6:	0 0 1 0 1 0 0 0	1 0 0 1 0 0 0 0

The preceeding information represents only a few of the many possible combinations in which you may configure your MMF Disk drive. As you can see, your MMF disk drive has capabilities far beyond those of other disk drives.

 ©1983, Micro Mainframe

2.1.0 PORT DESCRIPTION OF THE PCP-80 AND FPC-80

The MF-XX8X SERIES disk drive system for the Atari computer uses a powerful Z-80A CPU to send, receive, and assemble the information going to and from the Atari computer system. The Z-80 Computer architecture is substantially different from the 6502 CPU used in the Atari. The Z-80 has more internal registers than than the 6502 and the I/O may be either port mapped or memory mapped. For those of you who may be interested in learning more about the Z-80 CPU, we refer you to the Z-80 Technical Reference Manual (available from Zilog), as a discussion of these topics is beyond the scope of this manual.

Ports will be described in ascending numerical order. As used in this discussion, PCP-80 refers to the Z-80A CPU Card, FPC refers to the Floppy/Printer Controller, and Host Adapter refers either to the Z-80 Hot Clip/HA-3A combination or to the STD-HA-1. All products listed are designations of Micro Mainframe individual component parts.

1.1 HARD DISK HOST ADAPTER PORT USAGE

PORT 0C0H INPUT/OUTPUT

BIT	DESCRIPTION
ALL	Bi-Directional Data/Command, Completion Status

PORT 0C1H OUTPUT DEFINITIONS

BIT	DESCRIPTION
7	Output Bit Software Reset For Hard Disk Controller
6	Output Bit Select Hard Disk Controler
3	Output Bit Queue/Dequeue
0	Output Bit Controller ID

Queue mode is used when bits 6, 3, & 0 are all set during select. Dequeue is performed when bits 7, & 3 are set and output to Port (0C1H). PLEASE NOTE: **Queue & Dequeue are available only when used in conjunction with the Micro Mainframe DMX-1A Four Port Multiplexer.**

PORT 0C1H INPUT DEFINITIONS

BIT	DESCRIPTION
7	Not Used
6	Not Used
5	Not Used
4	I/O Direction Line; 0=Host to Controller; 1=Controller to Host
3	Busy Bit; 0=Not Busy, Not Selected; 1= Busy, Selected
2	MSG
1	Command/Data; 0=Data Bytes; 1=Command Bytes/Completion Status Bytes
0	Controller Data Request; 0=No Requests; 1=Requests

This concludes the Hard Disk Host Adapter Port Definitions. All remaining ports apply either to the PCP-80 CPU Module or to the FPC-80 Floppy/Printer Controller Module. We will make no attempt to relate them to the appropriate module at this time.

 ©1983, Micro Mainframe

2.1.2 DESCRIPTION OF PCP-80 AND FPC-80 PORTS

PORT 0E0H OUTPUT DEFINITIONS

BIT	DESCRIPTION
7	Enable All RAM (Disable ROM)
0	Interrupt Line to Atari (1=Interrupt)

PORT 0E4H OUTPUT DEFINITIONS

BIT	DESCRIPTION
7	Enable FDC Interrupts (1=Enable)
6	Enable Drive Timeout Interrupt (1=Enable)
5	Not Used
4	Drive Size Select (1=8" Drive; 0=5 1/4" Drive)
3	Not Used
2	Not Used
1	Drive Bank Select (0=Internal; 1=External)
0	Index Pulse Enable (0=Enable; 1=Disable)

PORT 0E4H INPUT DEFINITIONS

BIT	DESCRIPTION
7	FDC Interrupt Occurred (0); Did not Occur (1)
6	Drive Timeout Occurred (0); Did not Occur (1)

PORT 0ECH OUTPUT DEFINITIONS (OPTIONAL LED DISPLAY)

BIT	DESCRIPTION
ALL	Each Bit Corresponds to Each LED indicator

PORT 0ECH INPUT (DIP SWITCH SW1 - SEE PARAGRAPH 2.3.0)

BIT	DESCRIPTION
ALL	Each Bit Corresponds to Each Switch

PORT 0F0H OUTPUT DEFINITIONS (FDC COMMAND REGISTER)

BIT	DESCRIPTION
ALL	Write Command to Floppy Disk Controller Chip

PORT 0F0H INPUT DEFINITIONS (FDC STATUS REGISTER)

BIT	DESCRIPTION
ALL	Read Floppy Disk Controller Status

PORT 0F1H OUTPUT (FDC TRACK REGISTER)

BIT	DESCRIPTION
ALL	Write Track Number to Floppy Disk Controller

PORT 0F1H INPUT (FDC TRACK REGISTER)

BIT	DESCRIPTION
ALL	Read Track Number from Floppy Disk Controller

 ©1983, Micro Mainframe

PORT 0F2H OUTPUT (FDC SECTOR REGISTER)

BIT	DESCRIPTION
ALL	Write Sector Number to the Floppy Disk Controller

PORT 0F2H INPUT (FDC SECTOR REGISTER)

BIT	DESCRIPTION
ALL	Read Sector Number from Floppy Disk Controller

PORT 0F3H OUTPUT (FDC DATA REGISTER)

BIT	DESCRIPTION
ALL	Send Data to the Floppy Disk Controller

PORT 0F3H INPUT (FDC DATA REGISTER)

BIT	DESCRIPTION
ALL	Read Data from the Floppy Disk Controller

PORT 0F4H OUTPUT (DRIVE SELECT PORT)

BIT	DESCRIPTION
7	Density 0=Single; 1=Double
6	Wait on DRQ 0=Disable Wait; 1=Enable Wait on DRQ
5	Write Precomp 0=Disable Write Precomp; 1=Enable Write Precomp
4	Disk Side Number; 0=SIDE A; 1=SIDE B
3	Drive Select 3 (1) on Bank Bit 1 (PORT 0E4H)
2	Drive Select 2 (1) on Bank Bit 1 (PORT 0E4H)
1	Drive Select 1 (1) on Bank Bit 1 (PORT 0E4H)
0	Drive Select 0 (1) on Bank Bit 1 (PORT 0E4H)

PORT 0F4H INPUT (DIP SW 2 - SEE PARAGRAPH 3.0)

BIT	DESCRIPTION
ALL	Each Bit Corresponds to Each Switch

PORT 0F8H OUTPUT (LINE PRINTER 8 BIT DATA OUTPUT LATCH)

BIT	DESCRIPTION
ALL	Write a Byte to the Line Printer

PORT 0F8H INPUT (LINE PRINTER 4 BIT STATUS PORT)

BIT	DESCRIPTION
7	Printer Busy (1=Busy; 0=Not Busy)
6	Paper Out (1=Out; 0=Not Out)
5	Unit Select (1=Selected; 0=Not Selected)
4	Printer Fault (1=No Fault; 0=Fault Condition)
3	Not Used
2	Not Used
1	Not Used
0	Not Used

 ©1983, Micro Mainframe

PORT 0FFH OUTPUT (SERIAL DATA LINE TO ATARI)

BIT	DESCRIPTION
0	Serial Data Bit Out to Atari

PORT 0FFH INPUT (SERIAL DATA PORT AND CLOCK PORT)

BIT	DESCRIPTION
7	Serial Data Bit in from Atari
6	Serial Command Line from Atari
5	Not Used
4	Not Used
3	Not Used
2	Not Used
1	PCP-80 19.2K Baud Clock (used for PCP-80 Serial Write Clock)
0	Atari Clock In (Used for Serial Read Logic)

2.2.0 ROM SWITCH SW1 DEFINITIONS - NON-ATARI MODE

Two modes of operation are available when using the MF-XX8X series disk drives. These are the Atari Mode and the non-Atari mode. The Atari mode is the primary mode of operation and merely designates that the drive will emulate the Atari-810 disk drive with the MMF enhancements.

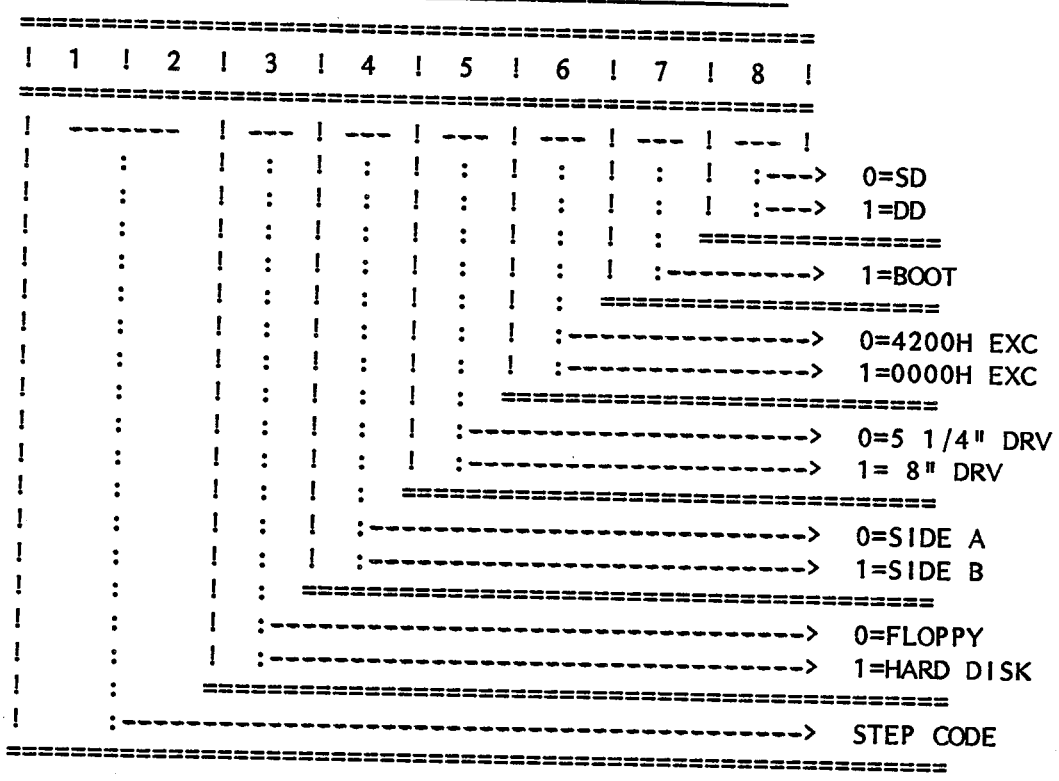
When specified, the non-Atari mode tells the drive to read the first sector on track 0 and load it into the PCP-80 RAM at the location specified by the position of SW1-6 (This will be either 0000H or 4200H). After this sector has been loaded program control will be transferred to the code located at that location. This feature allows the user to load operating systems such as CPM, OASIS, TRSDOS, or any other code and transfer control to the program. NOTE: **use of any of these modes will require additional hardware/software which is not included with the basic disk drive.**

Switch SW1 is used to select certain parameters necessary to operate your system. Switch SW1-7 must be ON (UP) when used in the Non-Atari Mode. When Switch SW1-7 is ON as noted above, the remainder of Switch SW1 must be configured for your particular installation according to the settings below. Any settings of SW2 are ignored when SW1-7 is ON (UP).

 ©1983, Micro Mainframe

NON-ATARI MODE SW1-7 ON (ON)

SWITCH SW1 CONFIGURATION



Where SD = Single Density
DD = Double Density
EXC= Execution Address
DRV= Drive

STEP CODE: The least significant two bits specify the step rate designators to be issued to the floppy disk controller during head movement operations. The setting of these switches will determine the speed at which the drives step from one track to another. Refer to the Western Digital specification manual for step rate information.

2.3.0 ROM SWITCH SW1 & SW2 DEFINITIONS - ATARI MODE

Switches SW1 and SW2 are used to select certain parameters necessary to operate your system in the Atari Mode. Switch SW1-7 must be OFF (Down) when used in the Atari Mode. When Switch SW1-7 is OFF as noted above, Switches SW1 and SW2 should be configured for your particular installation if you desire to operate in any mode besides the Atari Standard Mode (Single Density, No Line Printer, 5 1/4" disk drive, 1 drive available, and Base Drive Number = 1).

Please note that SW1-6 enables the printer when ON (UP). SW1 Switches 1-5 must be configured properly according to your particular installation.

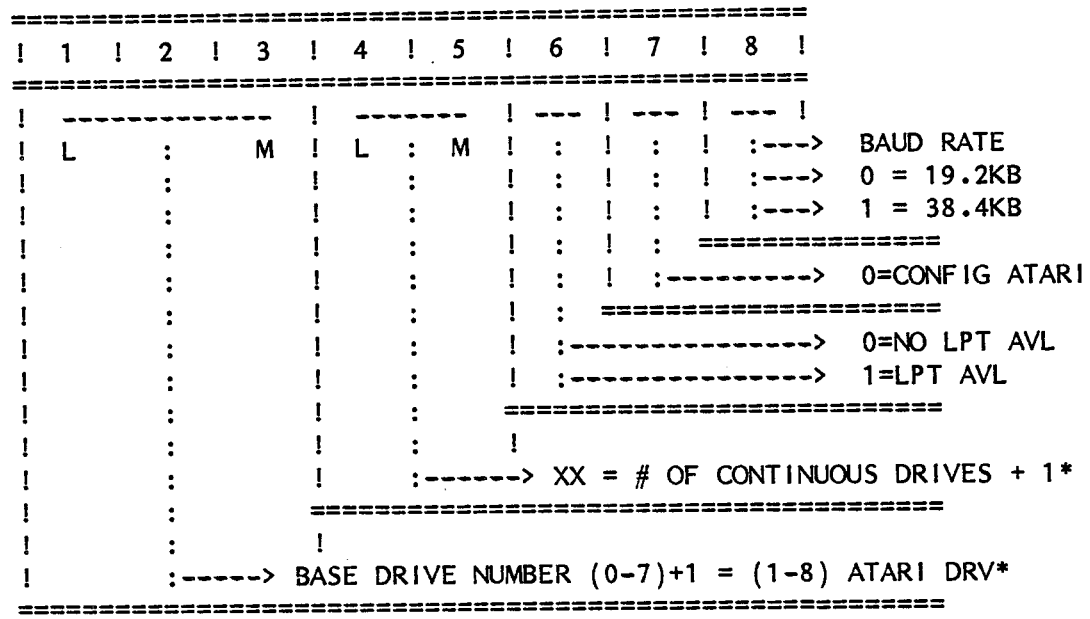
Switch SW1-8 when ON (UP) allows you to configure your MF-XX8X drive for 38.4 KB (double the standard Atari rate) when you are sending information from the drive to the Atari (this is valid only for future operating systems as none of the currently available systems are capable of supporting this function). The normal position of SW1-8 will be in the OFF (down) position. The drive is shipped from the factory with the following Default settings:

©1983, Micro Mainframe

SW1 All Switches will be initially set to OFF (DOWN).
SW2 All Switches will be initially set to OFF (DOWN).

These factory settings will allow your unit to emulate an Atari disk drive. To use the printer feature, double density, or 8" disk drives, you should configure the switches accordingly. **Please read paragraph 2.3.3 regarding the Programming Notes for SW1 & SW2.**

2.3.1 SWITCH SW1 CONFIGURATION - ATARI MODE



Where AVL = Available
DRV = Drive
STD = Standard Baud Rate
DBL = Double Baud Rate
* = Binary Coded

L = Least Significant
M = Most Significant

The first three switches program the BASE DRIVE NUMBER. The Base Drive is the number you desire to assign to the first MF-XX8X disk drive on the system. It is programmed starting with 0 and can be as high as 7. If, for example, you put the MF-XX8X drive as drive D1, then switches 1,2, & 3 will all be down or 0. If you have an Atari drive that you wish to install as D1 and a MMF drive that you wish to assign to drive D2, then lift SW1-1 up. Don't forget to press the reset switch on the MMF disk drive, or the disk drive will not read the change you just made. In similar fashion, drive D3 is programmed by lifting SW1-2 and pressing SW1-1 down. Drive D4 is programmed by pressing all three switches down and then lifting SW1-3. You should be able to figure out the remaining positions from the preceeding examples. Please remember that the switches add together in binary and that the leftmost digit is the least significant bit and the rightmost digit is the most significant bit.

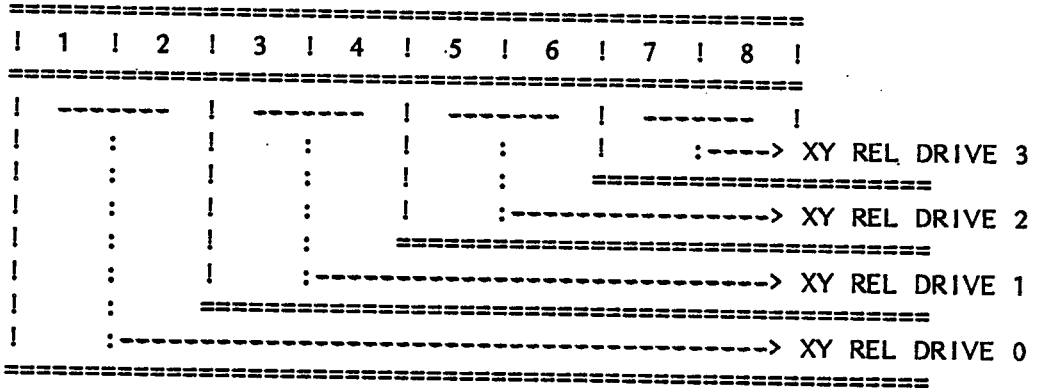
Switches SW1-4 and SW1-5 tell the MMF disk drive how many continuous drives are attached on the INTERNAL drive bus. Since only two bits of information are provided, it is easy to see that we may have 0, 1, 2, or 3 drives attached. Adding to these numbers, we find that we can have 1 through 4 disk drives that we can configure for the internal drive bus. Drives on the EXTERNAL drive bus must be configured into the device configuration table by the program ADDDRV which is

 ©1983, Micro Mainframe

included on your MMF distribution diskette supplied with the disk drive. Do not forget that any changes made to any of the switches on either SW1 or SW2 must be followed by pressing the reset switch on the MF-XX8X disk drive before the drive recognizes the new switch settings.

The remaining switches should be self explanatory. SW1-6 is used to tell the drive that the line printer is enabled when it is up, and disabled when it is down. SW1-7 tells the drive to configure itself as an Atari disk drive when it is in the down position. SW1-8 allows us to send the information to the Atari at 38.4KB instead of 19.2KB when it is up. This feature will work only when the Operating System has been rewritten to allow for this and currently is not supported by any vendor.

2.3.2 SWITCH SW2 CONFIGURATION - ATARI MODE



Where: X = Density of Drive (0=Single; 1=Double)
Y = Size of Drive (0=5 1/4"; 1=8")
REL = Relative

Please refer to XY Definition Below for map of definition.

Atari Drive Number = Base Drive Number + 1
Config for Drive = Relative Drive Switches

2.3.3 PROGRAMMING NOTES

Please note that switches SW1 & SW2 are programmed in binary notation. Paragraphs 2.3.1 & 2.3.2 both describe switches that are programmed in combination to obtain the desired action. These are SW1 (Bits 1-3), SW1 (Bits 4-5), SW2 (Bits 1-2), SW2 (Bits 3-4), SW2 (Bits 5-6), and SW2 (Bits 7-8). These switches have an L and an M noted under the switch drawing. These letters denote the Least Significant and the Most Significant bits in a BINARY configuration. You should become familiar with Binary notation before changing these switches. We will not attempt to give a complete tutorial on the binary system in this manual.

 ©1983, Micro Mainframe

2.4.0 XY DRIVE CONFIGURATION BIT MAP

The following is a description of the possible combinations of the drive configuration switches of SW2. Please note that these switches are used in pairs. Switches 1 & 2 are for Drive 1, Switches 3 & 4 are for Drive 2, Switches 5 & 6 are for Drive 3, and Switches 7 & 8 are for Drive 4.

XY SWITCH DEFINITIONS

00	5 1/4"	Single Density
10	5 1/4"	Double Density
01	8"	Single Density
11	8"	Double Density

 ©1983, Micro Mainframe

SECTION 3 - PROGRAMMING MANUAL

3.1.0 UTILITY COMMANDS SUPPORTED

The PROGRAMMING MANUAL describes some of the advanced features of the MF-168X and the MF-XX8X series disk drives for the Atari computer system by Micro Mainframe. Sample configuration procedures are given throughout. These advanced features will work only on MF-XX8X disk drives for the Atari and are not available for any other disk drive. THE LINE PRINTER SWITCH (SW1-6) MUST BE ON FOR THESE FEATURES TO WORK, SINCE THE INFORMATION IS TRANSMITTED TO THE MF-XX8X DISK DRIVE THROUGH THE LINE PRINTER CHANNEL.

In order to properly use this section of the manual, you will need the following: copies of the Western Digital Application notes and data sheets for both the WD1771 and WD1797 disk controllers; Atari Technical Reference Manual; Z-80 Technical Manual; and any Z-80 assembly language reference manual. These items are not available from Micro Mainframe.

You have probably realized by now that the Micro Mainframe Atari disk drive is more than just a simple disk drive. As shipped from the factory, the MF-XX8X is a powerful, 4 MHz. Z-80 computer system. A 64K memory expansion kit is available. A CPM package will be available in the future which will allow your Atari to operate as an inexpensive terminal for a more powerful computer system. This is all possible without giving up the basic function of performing as an Atari disk drive. Now you can have Atari compatibility when you desire, and switch to the CPM system when you want to take your computer beyond the limitations of the Atari system.

3.1.1 SOFTWARE COMMANDS SUPPORTED VIA THE LINE PRINTER CHANNEL

This series of commands allows the user to communicate to the PCP-80 computer and the FPC-80 disk controller of the disk drive. Micro Mainframe provides several utilities on the distribution diskette you received with your MF-XX8X disk drive. MMF presumes that you have a working knowledge of Z-80 programming and/or technical knowledge of disk operation in order to use some of these utilities. *** WARNING **** **If you do not have the required knowledge, we recommend that you become familiar with these procedures before using these commands. All consequences of improper use of any of these features become the liability of the user and Micro Mainframe will not be liable for any damage to any program or diskette caused through the mis-use of any of these commands.** Provision of these commands gives the knowledgable systems programmer a degree of control over his software never before available. One example of the use of these commands might be to create custom formatted diskettes for software distribution.

3.1.2 COMMAND FORMAT

The command format for transmitting a command to the PCP-80 portion of the Micro Mainframe Disk Drive is as follows:

27, 27, Op-Code, Auxiliary Data, Carriage Return

The PCP-80 computer will intercept any line printer data which has a leading sequence of ESC ESC (27 Decimal or 1B Hex), followed by a LEGAL op-code and auxiliary data. If only one ESC is found or the op-code is not supported, the PCP-80 will treat the data as if it is line printer data. If the PCP-80 was powered up with the line printer configuration switch (SW1-6) in the off or unavailable setting, the PCP-80 will not respond to any line printer channel commands. To issue commands when the above is true, use the disk command programming method.

When two sequential ESC characters are detected, the PCP-80 will examine the following byte to determine if it is a valid op-code. If it is valid, the routine

 ©1983, Micro Mainframe

appropriate to execute your command is entered. If the op-code is illegal, the data is treated as line printer data.

The PCP-80 has 3 predefined device handlers. These handlers are numbered 0-2 and their functions are as follows:

HANDLER NUMBER	DESCRIPTION
00	Floppy Disk Device Handler
01	Line Printer Handler
02	Hard Disk Handler

3.1.3 OP-CODES SUPPORTED

The following OP-CODES are currently supported by the PCP-80.

OP-CODE	DESCRIPTION
00	Define a Device.
01	Remove a Device.
02	See Note.
03	Dump PCP-80 RAM to Line Printer.
04	Modify PCP-80 RAM contents.
05	Read a track from the floppy disk.
06	Burn a track on the floppy disk.
07	Issue commands & data directly to the hard disk controller.
08	Software reset for the PCP-80 CPU.

3.1.4 OP CODE DESCRIPTIONS

The auxiliary data bytes associated with the various OP CODES are defined as noted below. You are referred to the Atari Technical Reference Manual for further information on the manner in which the following are used.

RELATIVE BYTE	DESCRIPTION
00	Atari Serial Device ID, (For Drives 31Hex-38Hex).
01	Handler Number (0-N) see above. (Normally 00 for Floppy Drive)
02	Floppy drive type. See Relative Byte 2 below.
03	Number of cylinders for this drive.
04	Number of sectors per track for this drive.
05	Number of heads for this drive (0-1 for floppy).
06	Atari Command Status (see below).
07	Hardware status (used by Atari).
08	Device time-out value (used by Atari).
09	CRC byte of status used by Atari.

 ©1983, Micro Mainframe

RELATIVE BYTE 02 HAS A BIT MAPPED DEFINITION AS DEFINED BELOW

BIT	DEFINITION
7	Density of Drive (0=Single; 1=Double)
6	Not used
5	Not used
4	Size of Disk (0=5 1/4"; 1=8")
3	Internal Use
2	Not used
1-0	Step Rate for Drive

EXPANSION OF STEP RATE BITS

BITS 0-1	5 1/4"	8"
00	6 MS	3 MS
01	12 MS	6 MS
02	20 MS	10 MS
03	30 MS	15 MS

Relative byte 06 must have bit 5 set for double density and reset for single density. Other bits will be used internally by the PCP-80.

NOTE 1: The number of heads should be the physical number of heads on the drive minus one. For a double headed drive, the number of heads should be set to 1. The number of sectors per track (Relative byte 04) should equal the number of sectors per side per track. The PCP-80 will automatically double the number of sectors per track when two sided operation is specified.

NOTE 2 (see NOTE 1 also): The PCP-80 is designed to use 18 sectors per track for 5 1/4" operation and 26 sectors per track for 8" operation. The PCP-80 will double the number of sectors per track for double headed operation as mentioned above.

The PCP-80 supports any number of tracks per side up to a maximum of 255 tracks, thus allowing for future high capacity floppy disk drives.

3.2.0 PROGRAMMING EXAMPLES

The following program is a sample Basic program that allows you to send a command line to your PCP-80 to configure drive number D2. NOTE: This will only work on drives that are controlled directly by the PCP-80 and will not work on your Atari or Percom drive.

EXAMPLE 1 - SINGLE DENSITY, SINGLE SIDED, 40 TRACKS, 18 SECTORS PER TRACK, AND 6 MS STEP RATE.

```
10  DATA 27, 27, 0, 50, 0, 0, 40, 18, 0, 0, 0, 255, 0, 0
20  DIM A$ (14)
30  FOR G = 1 TO 14 : READ A : A$ (G,G) = CHR$ (A) : NEXT G
40  LPRINT A$
50  END
```

 ©1983, Micro Mainframe

EXAMPLE 2 - DOUBLE DENSITY, DOUBLE SIDED, 40 TRACKS, 18 SECTORS PER TRACK, AND 6 MS STEP RATE.

```
10  DATA 27, 27, 0, 50, 0, 128, 40, 18, 1, 32, 0, 255, 0, 0
20  DIM A$ (14)
30  FOR G = 1 TO 14 : READ A : A$ (G,G) = CHR$ (A) : NEXT G
40  LPRINT A$
50  END
```

This concludes the discussion on configuration methods for different types of floppy disk drives, and should be self explanatory for other types of devices if you read carefully the definitions and references noted above.

3.3.0 LINE PRINTER DEVICE DESCRIPTION BLOCK

This device will automatically be included in the configuration table if the line printer available switch is set on power-on or reset, otherwise it is not included.

BYTE	DESCRIPTION
00	DEVICE ID. (Normally 01 for Printer)
01	Handler Number (Normally 01 - Printer)
02	Not Used
03	Not Used
04	Not Used
05	Not Used
06	Command Status
07	Last line length byte (Atari definition)
08	Time-out
09	Not Used
10	CRC byte of Status Request

3.4.0 HARD DISK CONTROL BLOCK

Although the hard disk control block seems complex, this type of structure allows it to be quite flexible. We will not attempt to give every configuration possible. Instead, we refer you to the MMF-Atari Hard Disk Manual shipped with your MMF hard disk system.

3.4.1 HARD DISK DEVICE BLOCK

BYTE	DESCRIPTION
00	Device ID
01	Handler Number (02)
02	Base LUN and Step Rate (see below)
03	Low Byte of Sector Offset to start of disk area
04	High Byte of Sector Offset to start of disk area
05	Hard Disk, Controller Type, MSB of sector offset to start of area (see below)
06	Command Status (bit 5 should always be set for 815)
07	Hardware Status
08	Time Out
09	Not/Used
10	CRC bytes of Status

 ©1983, Micro Mainframe

3.4.2 BIT MAP OF RELATIVE BYTE 02

BASE LUN AND STEP RATE FIELD

BIT	DESCRIPTION
7-6	Ignored by DTC controller Step Rate for Xebec controller. Defined as follows: 00 = Standard 3 MS step rate 01 = Half Step Option (Seagate, Texas Inst) 10 = Half Step Option (Tandon) 11 = Buffered step mode (CMI, RMS, Seagate 412: 200 MS)

The step rate information will be translated by the PCP-80 to the following and placed into the control field bits (2-0):

00 = 000
01 = 001
10 = 010
11 = 100

5-0	Value used to determine base LUN number ID-(Base LUN) = physical hard drive number (0-7)

3.4.3 BIT MAP OF RELATIVE BYTE 05

HARD DISK, CONTROLLER TYPE, MSB OF OFFSET

BIT	DESCRIPTION
7	Always set to indicate a hard disk.
6-5	Controller Type. Defined are: 00 = XEBEC 01 = DTC 10 = Not Defined 11 = Not Defined
4-0	MSB of Base Offset to start of area

The Offset to the start of the disk area is defined as **The Number of sectors offset from 0 to the start of the disk area for this device.** A total of 21 bits has been allocated to the offset sector number. This number is then added to the sector number desired to achieve the sector number on the disk that we wish to address. This allows either for the hard disk to be partitioned in various ways or no partitioning at all. The maximum size of a single file is therefore limited to a maximum of 65536 sectors per device partition excluding the directory or other system information. This is not a limitation of the Micro Mainframe unit, but is due to the two-byte sector addressing method used by Atari. The 21 bits allocated for sector offset gives the Micro Mainframe unit a maximum capacity of 2,097,150 sectors which may vary in length depending upon the hardware used.

 ©1983, Micro Mainframe

3.4.4 OP-CODE DEFINITIONS

We shall briefly discuss each OP-CODE and give an example in Basic that should illustrate the use of each op-code.

OP CODE 00 - DEFINE A DEVICE

FORMAT:

27,27,0,DEVICE ID,HANDLER #,DRIVE TYPE,NO. CYL,NO. SECT/TRK,NO. HEADS,CMD STATUS,HARDW STAT,TIME OUT,CRC OF STAT

Each of the devices supported by the PCP-80 has different device blocks. Any device supported by the PCP-80 will have a block of data exactly 12 bytes long which contains information as to which device ID, handler, device parameters, device time out information, and the status information needed by the Atari computer.

We must first send the double escape sequence, then the OP CODE for the operation we desire, and finally the device parameters required by the device. **PLEASE NOTE: the Micro Mainframe drive follows the protocall established in the Atari DOS manual to define devices. You MUST read your Atari DOS Manual Appendix G in order to understand how to use this feature of the MMF disk drive.**

When specifying a device, please remember that you are working with decimal numbers when you are in Basic, and Hex numbers when using Assembly language.

For example, the device ID for floppy disk drives begins with 49 for Drive D1 and ends with 56 for Drive D8:. The Hex numbers that correspond to those numbers are 31H for drive D1: and 38H for Drive D8:. Two equivalent methods are given below to specify in Basic the device ID for drive D5:, and you may use either method.

LPRINT CHR$(53) IS EQUAL TO LPRINT"5"

```
10 LPRINT CHR$(27);CHR$(27);CHR$(0);CHR$(53);CHR$(0);
CHR$(0);CHR$(40);CHR$(18),CHR$(0),CHR$(0),CHR$(0),CHR$(255);
CHR$(0);CHR$(0)
```

The preceeding Basic statement causes the MMF drive define drive D5: as follows:

Use handler No. 00, Drive D3, Single Density 5 1/4" drive, 40 cylinders, 18 sectors per track, single sided, hardware status 255, device timeout 0, and CRC byte 0.

PLEASE NOTE: After you have defined a device, you must hit the SYSTEM RESET key on your Atari in order to tell the MMF drive to get the updated drive status.

OP-CODE 01 - REMOVE A DEVICE

FORMAT:

27, 27, 01, (SERIAL DEVICE ID)

This command will remove the device number specified if the device is defined or ignore the command if the device is not currently defined. If the device exists, it is consequently removed from the Device Definition table is reduced in size by 12 bytes.

10 LPRINT CHR$(27); CHR$(27); CHR$(1); "4" or CHR$(52)

This command will remove drive number 4 when RUN is typed.

 ©1983, Micro Mainframe

OP-CODE 02 - RESERVED FOR SYSTEM USE
OP-CODE 03 - DUMP PCP-80 RAM

FORMAT:

27, 27, 3, ADD LOW, ADD HI, LEN LOW, LEN HI

This command will cause the PCP-80 computer to DUMP the contents of its RAM to the line printer. The auxiliary data is the RAM Address Low Byte, RAM Address High Byte, Length Low Byte, and Length High Byte. Try this example:

```
10  LPRINT CHR$ (27); CHR$ (27); CHR$(3); CHR$ (0); CHR$(64); CHR$(0); CHR$ (1)
```

This example will send a Hex dump of the PCP-80 RAM to the line printer. Ram will be dumped from address 4000 Hex to 40FF Hex, and 1 page of RAM (256 Bytes each page) will be sent to the line printer. To use this example, you must have a line printer connected to the MF-XX8X and it must be properly enabled.

OP CODE 04 - MODIFY RAM

FORMAT:

27, 27, 4, ADD LOW, ADD HIGH, 0 < LEN < = 32, DATA

This example will place the data bytes into the PCP-80 RAM beginning at the address specified for the length specified. Use this command with caution, as you might cause the PCP-80 to malfunction. Any use of this function is at your own risk, but we have included it for those who have the knowledge to make it function properly (ie. - knowledge of Z-80 programming and how disk systems operate internally).

```
10 LPRINT CHR$(27); CHR$(27); CHR$(4); CHR$(0); CHR$(64); CHR$(3); CHR$(0); CHR$(255); CHR$(128)
```

This example will place the hex values 00, FF, and 80 into the PCP-80 RAM beginning at address 4000H.

OP-CODE 05 - READ TRACK

FORMAT

27, 27, 5, SERIAL DEVICE ID, TRACK #

Issuing this OP-CODE will cause the PCP-80 to read a track from the floppy disk drive (without regard to the type of data present) and send it to the line printer as a Hex dump. This function will allow you to determine how any track is organized on the diskette, and will print all of the information regarding sector format.

```
10 LPRINT CHR$(27); CHR$(27); CHR$(5); "1"; CHR$(1)
```

This command will cause Track 1 of Drive D1 to print out on the line printer (the line printer must be attached and enabled as per paragraph 3.0 of the operations manual before it will work). This routine might be useful to determine the number of sectors on that track, the numbering scheme of the sectors on that track, etc.

 ©1983, Micro Mainframe

OP-CODE 06 - BURN TRACK

FORMAT

27, 27, 6, SERIAL DEVICE ID, TRACK #

```
10 LPRINT CHR$(27); CHR$(27); CHR$(6);"1"; CHR$(1)
```

The example above will cause the PCP-80 to destroy track 1 on drive D1. The track that you wish to destroy must have been previously formatted before you issue the command. Since this command makes the track unreadable, we advise that you use extreme caution when using it as it could render your diskette useless. You may only use this command with the floppy disk, as it is inoperative when used in conjunction with the hard disk. The example for this command is identical to that for OP CODE 4 as noted above.

OP-CODE 07 - ISSUE COMMANDS TO THE HARD DISK CONTROLLER

FORMAT:

27, 27, 7, DIR, 0 > LEN < = 30, CMDBLK, DATA

This OP-CODE provides you with the capability of issuing commands directly to the hard disk controller.

The firmware for the hard disk has been designed for maximum flexibility. As such, it provides a step-rate field which is used by Xebec controllers for several different step rates supported by that controller. The Base LUN # is used in conjunction with the device ID in order to allow multiple hard disks to be attached and selected automatically. The sector offset field is to be used for device partitioning. The hard disk type flag and controller type bits are provided to allow the user to configure for almost any SASI-BUS hard disk system. The Base LUN # Field (Byte 3, Bits 5-0) is subtracted from the device ID to achieve the physical LUN # to be addressed (where LUN = Logical Unit Number). For example, a Device ID # of 50 Hex and a Base Unit # of 49 Hex equals a LUN # of 1 (50 - 49 = 1).

Sector Offset Field bytes 4, 5, (6 <Bits 0 - 4>) represent the number of sectors offset from the start of the disk to begin this new area on the disk. The number thus obtained is added to the desired sector number to obtain the physical sector number.

One could, for example, configure the PCP-80 to handle the Xebec controller for a 10 or 15 megabyte drive. This same command could also allow you to configure for other SASI-BUS* compatible controllers or different drive sizes. An explanation of the format follows:

```
    DIR = Direction of the activity; 0=READ; 1=WRITE.
    LEN = Number of data bytes supplied (<=30 Supplied).
          The routine will supply more if needed but the user
          supplied byte count is limited to 30.
CMDBLK = Hard disk controller's command block (consult the
          appropriate manual for the controller used).
   DATA = Any data required for the operation requesting.
SASI-BUS  = Shugart Associates Standard Interface Bus
```

Please consult the manual appropriate for your hard disk for examples of the programming methods.

 ©1983, Micro Mainframe

OP-CODE 08 - RESET

FORMAT

27, 27, 8

This command will cause the PCP-80 to restart execution of the firmware at the beginning of the code (effectively, a power on reset). If the controller is locked up due to a hardware condition, this command will have no effect and you will have to turn the power on the disk drive off and then back on to achieve a reset condition.

3.5.0 LOGICAL ADDRESS COMPUTATION

The following information is provided for those knowledgable in the methods for performing disk I/O on both floppy disk drives and hard disk drives. Not everyone will understand the purpose of this information, but it is provided for those who have a need to know.

3.5.1 FLOPPY DISK LOGICAL ADDRESS COMPUTATION

ATARI SECTOR # = (CYADD + HEAD) * SECTRK + SECTOR

Where:

CYADD = CYLINDER # (0-N) DESIRED
SECTOR = SECTOR NUMBER DESIRED
HEAD = NUMBER OF HEADS PER CYLINDER (0-1)
SECTRK = NUMBER OF SECTORS PER TRACK (ONE SIDE)
PTRK = PHYSICAL TRACK

Then:

PTRK # = INT(ATARI SECTOR / (SECTRK * (# HEADS + 1)))
SECTOR NUMBER = ATARI SECTOR - (PTRK # * (SECTRK * (# HEADS + 1)))
SIDE = IF SECTOR # > SECTRK THEN (SIDE B)

3.5.2 HARD DISK LOGICAL ADDRESS COMPUTATION

LOGICAL ADR = (CYADR * HDCYL + HDADR) * SECTRK + SECADR

Where:

CYADR = CYLINDER ADDRESS
HDADR = HEAD ADDRESS
SEADR = SECTOR ADDRESS
HDCYL = NUMBER OF HEADS PER CYLINDER
SECTRK = NUMBER OF SECTORS PER TRACK

 ©1983, Micro Mainframe

APPENDIX A

A.1.0 APPLICATION NOTE - 1771/1797 DISK CONTROLLER

The Micro Mainframe FPC (Floppy/Printer Controller) uses the Western Digital 1797 Floppy disk controller chip (FDC) where the Atari 810 disk drive uses the Western Digital 1771 FDC. There are several differences between the 1771 and 1797 that are important for the user to understand.

The WD1771 operates in the FM (Frequency Modulation) mode and is capable only of single density operation. The WD1797 is capable of both FM and MFM (Modified FM) operation and can therefore read and write double density information as well as single density information. Both of these chips perform the same function (ie. - to read and write information to the diskette) but the WD1771 has the capability to write 4 different Data Address Marker (DAM) Bytes. The WD1797 has the capability to write only 2 of the 4 possible DAM bytes. Although the WD1797 has the ability to read all 4 of the different DAM bytes, it can report only two of them. It is this difference that can create compatibility problems between diskettes written by the WD1771 which are to be read on systems using the WD1797. The following chart will illustrate the possible combinations of DAM bytes for both chips.

1771	1797	MARKER NAME
F8	F8	Deleted Data Marker
F9	XX	User Defined
FA	XX	User Defined
FB	FB	Data Marker

Note: XX denotes not supported.

As previously stated, the WD1797 can read the sector data written with any data marker but, is unable to report the two user defined types in the completion byte.

The DAM byte inconsistency between the WD1771 and WD1797 could, conceivably, cause some software compatibility problems when copy protected software uses one of the unsupported DAM bytes. **Since the normal Atari 810 disk drive uses only the FB data marker, there should be no problems in this area.** Third party software manufacturers have been informed of this potential problem area and should restrict their data marker usage to the data markers supported (ie. - FB and F8). Fortunately for the consumer, most vendors are using only the data markers supported by both chips. This will soon become a necessity once Atari releases their disk drive which uses the WD2793 controller chip. This chip, like the WD1797 is capable of reporting only the F8 and FB data markers. It is the position of Micro Mainframe that software vendors who insist upon using the F9 and FA data markers in production software will not be compatible with Atari when they release their new disk drive later in 1983.

In summary, both the WD1771 and WD1797 can read the data bytes produced by either controller as long as it is single density (FM mode); the WD1797, however, is capable of reporting only two of the four data markers that the WD1771 can produce. The WD1771 is capable of single density operation only whereas the WD1797 is capable of both single and double density operation. As far as compatibility is concerned, the WD1797 is completely compatible in regards to the needs of the Atari disk drive. If third party software vendors elect to incorporate the F9 or FA data markers in their protection scheme, there could exist some degree of incompatibility with the Micro Mainframe disk drive as well as Atari's new disk drive and any others using double density controller chips.

 ©1983, Micro Mainframe

APPENDIX B

B.1.0 APPLICATION NOTE - BAUD RATE SELECT

Micro Mainframe has incorporated within the firmware package of the MF-XX8X, a softswitch RAM location to signal to the MF-XX8X the write serial bit baud rate. The RAM location for this softswitch is at 404DH. The switch may be changed at any time and will effect only the write baud rate, NOT the receive baud rate as the MF-XX8X will track any reasonable baud rate up to 76.8k baud and possibly beyond.

There are two ways of changing the write baud rate. The first is to select either 19.2k or 38.4k by the dip switch SW1-8. When the switch is DOWN or OPEN the baud rate is 19.2 KB, the standard Atari baud. When the switch is UP or CLOSED the baud rate is 38.4 KB. Please note that switch SW1-8 is read only during initial power on or a manual reset condition which is supplied by pressing switch SW-3 on the back panel of the MF-XX8X disk drive. The value thus obtained is stored in the softswitch RAM location 404DH.

The second method of changing the baud rate is through the MOD RAM command through the line printer channel using OP-CODE 04 (please refer to Section III of the Technical Manual). The following BASIC statments demonstrate how to change the baud rate using this method. Please note that this feature will not be valid until it is supported by some vendor or vendors of operating systems. Inclusion of this feature is just one example of Micro Mainframe's continuing dedication to provide you with advanced features unobtainable through any other manufacturer.

TO SET 38.4 K BAUD:

```
10 LPRINT CHR$(27);CHR$(27);CHR$(4);CHR$(77);CHR$(64); CHR$(1);CHR$(255)
```

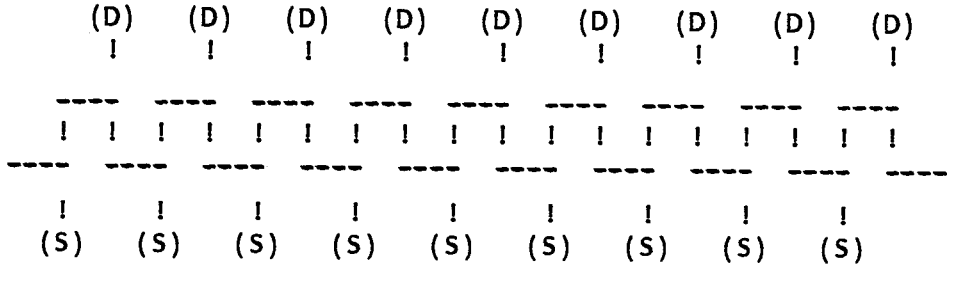
TO SET 19.2 K BAUD:

```
10 LPRINT CHR$(27);CHR$(27);CHR$(4);CHR$(77);CHR$(64); CHR$(1);CHR$(0)
```

Please remember that the read clock will track the Baud rate sent to the Atari regardless of the setting of the of the write clock baud rate.

B.1.1 19.2 K BAUD CLOCK WAVEFORM

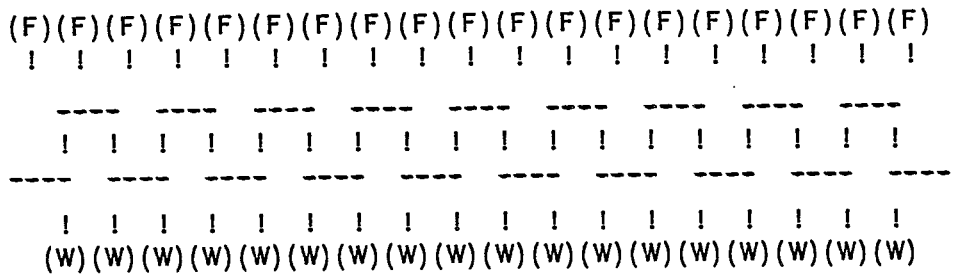
The MF-XX8X uses a true square wave clock with a 50% duty cycle. This clock is used to control the transmitting of serial data bits between the Atari and the MF-XX8X. The diagrams below are conceptual drawings of the clock rates for both the 19.2 KB and 38.4 KB waveforms.



Each rising edge of the clock marks the (S)tart of a data bit window. Each falling edge represents the the location of the data bit or when the data (D) bit is valid.

©1983, Micro Mainframe

B.2 38.4 K BAUD CLOCK WAVEFORM



During 38.4k baud transmission, each transition of the clock marks a data bit write window (W). The (F)ast baud window mark (W) occurs in the middle of every high or low condition of the clock.

 ©1983, Micro Mainframe

APPENDIX C

C.1.0 UTILITY PROGRAMS SUPPLIED BY MICRO MAINFRAME

Micro Mainframe has supplied several utility programs on a single density diskette that may be useful to you. Of these, CONVERT.V1, DISASSEM.V1, and MICROZAP are written in assembly language. The other programs are written in Atari Basic and you must have your cartridge installed. PLEASE NOTE THAT SOME OF THESE PROGRAMS REQUIRE THAT YOU CONFIGURE SW1-6 FOR LINE PRINTER OPERATION WHETHER OR NOT YOU HAVE A LINE PRINTER. IF YOU DO NOT HAVE A LINE PRINTER, PLEASE MAKE CERTAIN THAT YOU RECONFIGURE SW1-6 ACCORDINGLY WHEN YOU FINISH OR THE SYSTEM MAY HANG UP.

Please make a backup copy of any diskettes you intend to use with any of the following programs. If you tell the MMF drive to do something that you do not really want to do, your programs may be destroyed. Some of the programs supplied are written in Atari Basic. These programs actually transfer a Z-80 machine language program stored in Basic DATA statements to the computer inside the MMF disk drive. Once the program has transferred into the drive, you will get the message STOPPED IN LINE 310. This does not mean that the program has stopped, but that the Atari has transferred control over to the MMF disk drive.

Most of the programs have been configured for drive D2: to begin with. We have set the programs up in this manner so that you will not destroy something that you did not wish to overwrite. If you have a single drive system, change the first 32 in Data statement 1070 to a 31. One such example is given below in paragraph C.1.3.

C.1.1 CONVERT.V1

CONVERT.V1 is a machine language program designed to move files from one diskette to another regardless of density. It does not require you to alter of any of the drive switch settings, as it automatically configures according to the density of the diskette. IT IS NOT NECESSARY TO HAVE THE LINE PRINTER SWITCH UP TO USE CONVERT.

To CONVERT from one format to another, remove your Basic Cartridge and type L from DOS. Insert the Utility Diskette supplied by Micro Mainframe into the drive and type CONVERT.V1. The program will load and after displaying the banner, it will give the following display:

FILE SPEC. / MASK ?

Your response may take any of the following forms:

EXAMPLES OF THE WILD CARD OPTIONS

1. ATAR.BAS <CR>
2. *.* <CR>
3. *.* Q <CR>
4. A*.* <CR>
5. *.BAS <CR>
6. A*.* Q <CR>
7. *.BAS Q <CR>
8. ATAR*.* <CR>
9. AT??.* <CR>

 ©1983, Micro Mainframe

If your response is a carriage <RETURN> with no other characters input, the program will type out:

INSERT SYSTEM DISKETTE AND PRESS <START>

Pressing the <START> button will take you back to DOS. This is the method used to exit the program when you are through copying from a diskette, or you may elect to copy some additional files from another diskette.

OPTIONS ALLOWED

The * represents a wild card wherein CONVERT does not care what letter is after that position; The ? is a single character mask; The Q tells CONVERT that you wish to be querried regarding the copy status of each file. PLEASE NOTE: You must insert a space between the last character or mask character of the file name and the Q or CONVERT will consider the Q as part of the file name.

In example (1) above, only the file named ATAR.BAS will be copied. Example (2) will instruct CONVERT to copy any file with or without any extension (denoted by *.*). Example (3) will perform the same function except that it will Query the user regarding each file in the directory. If the response is Y, the file will be copied; If it is N or anything else, CONVERT will bypass that file and query regarding the next file.

Example (4) will copy all files beginning with A. Example (5) will copy all files with a .BAS extension. Example (6) will copy all files beginning with A and will Query if you wish to include that file. Example (8) will copy all files that have ATAR as part of the file name. Example (9) will copy all files that begin with AT and any two additional characters, and with any extension.

We have attempted to give you some of the possible combinations that you may use with CONVERT, but these are by no means the only form of the mask possibilities. The following is a sample session using CONVERT.

Load the CONVERT program. When CONVERT asks,

FILESPEC / MASK ?

Specify *.* to tell CONVERT to convert all files on the source diskette to double density. CONVERT will then respond:

INSERT SOURCE DISKETTE AND PRESS <START>

It will then display the selected files and the following message:

FILE TO COPY : D1:XXXXXXXX

where XXXXXXXX is the name of the first file in the list of selected files. If you specified the Query option, CONVERT will respond ENTER 'Y' TO CONVERT. If you desire to copy that file respond with Y, otherwise, respond with an N. If you did not specify the Query option, CONVERT will proceed with the conversion. CONVERT will then load the file from the source diskette and tell you INSERT DESTINATION DISKETTE AND PRESS <START>. Please follow the prompts on the screen to complete the conversion process. When you are through with the conversion and wish to exit the program, hit <RETURN>, insert the system diskette when prompted, and hit <START>. This will return you to DOS.

 ©1983, Micro Mainframe

C.1.2 FASTCOPY.V11

This program is a BASIC program that will copy any NON-PROTECTED diskette to another formatted diskette in about 45 seconds. PLEASE NOTE: USE OF THIS PROGRAM REQUIRES A DUAL DRIVE MICRO MAINFRAME SYSTEM TO WORK. CONVERT WILL NOT WORK WITH AN ATARI OR PERCOM DRIVE AS ONE OF THE DRIVES. IT WILL WORK ONLY WHEN BOTH DISK DRIVES ARE CONTROLLED BY THE MICRO MAINFRAME DISK CONTROLLER. YOU MUST HAVE THE LINE PRINTER SWITCH UP TO USE THIS PROGRAM.

FASTCOPY will not work on unformatted diskettes. You must format a diskette before you proceed with it. If you do not, the program will take no action and abort.

The user may change line 1070 in these programs to make minor changes to reconfigure for different source and destination drives. Please note that this is not the entire program. Users should feel free to rewrite any portion of any of the programs supplied to accomplish their own special purposes. MICRO MAINFRAME CANNOT ASSIST YOU WITH THESE EFFORTS. These programs are given as an example of how to use the down load capability of the Micro Mainframe disk drive and to demonstrate the speed of the Micro Mainframe system.

```
!------ SOURCE DRIVE NUMBER
! !---- DESTINATION DRIVE NUMBER
! !
```

1070 DATA 313200000000F3 ..... ETC.

Please note that the data must be typed in Hex and must not have any spaces between the numbers. If, for example, you desired drive D2 to be the source and drive D1 to be the destination, you would change the listing to look like the following example.

1070 DATA 32310000 ..... ETC.

Notice that the code for the source and destination drives has been reversed. The code used to represent the drive specification in the preceeding data statements is the ASCII representation of the physical number of the drive desired. PLEASE NOTE THAT IN ALL OF THE BASIC PROGRAMS ON YOUR RELEASE DISKETTE, DRIVE D2 HAS BEEN SPECIFIED IN LINE 1070 OF PROGRAMS AS ORIGINALLY RELEASED. THIS WAS DONE TO KEEP YOU FROM ACCIDENTALLY DAMAGING A DISKETTE IN DRIVE D1. YOU WILL HAVE TO CHANGE THE VALUE TO REFLECT THE APPROPRIATE DISK DRIVE THAT YOU WISH TO USE. REMEMBER THAT THESE PROGRAMS ACTUALLY TRANSFER A Z-80 MACHINE LANGUAGE PROGRAM TO THE COMPUTER IN THE MMF DISK DRIVE TO PERFORM THE WORK. WHEN YOU SEE A MESSAGE LIKE STOPPED IN LINE 320, YOU WILL KNOW THAT WORK IS BEING PERFORMED BY THE DISK DRIVE'S ONBOARD COMPUTER AND THE RED LED ON THE DISK DRIVE SHOULD ILLUMINATE FOR SEVERAL OF THE PROGRAMS INCLUDED WITH YOUR DISK DRIVE.

To designate a specific disk drive, you must use the ASCII equivalent number to specify the drive number. For example: Drive 1 = 31, Drive 2 = 32, Drive 3 = 33, and Drive 8 = 38. All drives in between will follow logically throughout the sequence.

 ©1983, Micro Mainframe

C.1.3 PRTSEC.V11 - PRINT SECTORS

This program is a Basic Program that will allow you to print out a listing of all of the sectors found on the diskette located in the drive specified. THE MF-XX8X MUST BE CONFIGURED FOR THE LINE PRINTER AND THE PRINTER MUST BE ATTACHED FOR THIS PROGRAM TO OPERATE. If the DAM byte is not an FB Hex, PRTSEC will print F8 Hex beside each sector that has that DAM byte present. REMEMBER, that the 1797 controller will not be able to report DAM bytes of F9 or FA Hex.

```
          ,------- DRIVE SPEC
          !
          !
1070 DATA 32000000 ..... ETC.
```

C.1.4 PRTTRK.V11 - PRINT TRACKS

This program is identical to PRTSEC except that it will print out a listing of all of the tracks found on the diskette. The configuration is identical to that of PRTSEC above. The program will list out all of the tracks found on the diskette.

C.1.5 SETDAM.V11 - CHANGE DAM BYTE

This program allows you to change the Data Address Marker (DAM) byte for any sector number. It functions by reading a sector into the memory of the PCP-80, changes the DAM byte to the one that you have specified, and writes it back out to the diskette in the proper location. THE LINE PRINTER SWITCH MUST BE ON AND PROPERLY CONFIGURED FOR THIS ROUTINE TO OPERATE. It is not necessary to have a line printer attached for this program to operate. There are two lines that the user may desire to change in this program. These are listed below along with an explanation of information you may wish to change.

```
,------------- DRIVE #
!     ,------- TRACK #
!     !  ,---- SECTOR #
!     !  !
!     !  !
1070 DATA 3100002709 ..... ETC.
```

The VALUE circled in line 1100 of the program listing is the OP-CODE that tells the WD controller to write sector 9 of Track 27H using the F8H Deleted Data Marker. If you wish to write the FBH Data Marker instead, you should change this byte to A8. As noted previously, these are the only two DAM bytes supported.

1100 DATA 5B31403A8B404FDD46037BD3EC3E (A9) CDB807E6FCC9

We have left some spaces in the listing to highlight the command given to the WD controller so that you may see its position in the line of code. Please make certain that there are no spaces in your lines when you modify any of the codes described above. If you leave a space in the line, the program will have a 20 Hex in that location which is the ASCII representation for a space code. Needless to say, your results could be unpredictable.

 ©1983, Micro Mainframe

C.1.6 DISASSEM.V1

This program is a machine language disassembler that allows you the option of disassembling a disk file, Atari memory, or the boot sector of any diskette. IF YOU DESIRE THE LISTING TO GO TO THE PRINTER, YOU MUST HAVE THE LINE PRINTER SWITCH UP. The program is self prompting and requires only the following information:

1. You will be prompted regarding whether you wish the listing to go to the line printer. Your response should be Y or N.

2. You will be asked if you desire the cross reference table, which is a useful item to have when you are trying to learn how a program is written as it will help you follow the logic flow.

3. You will be asked if you desire the cross reference table to be printed on the line printer.

4. You will be given the opportunity to specify a RESTART LOCATION if desired. This is the location at which you desire the printing of the disassembly to start. Regardless of the location you specify, the complete disassembly will be performed, but the listing (to either video or printer) will begin at the location specified. If you want the entire listing, you should hit <RETURN> to this prompt. This feature is useful when you only need to see a portion of the code and you do not have to spend the time to wait until the desired portion has been performed.

5. You will be asked for a FILESPEC. You have three options at this point:

   a. You may type in the name of the file and DISASSEM will proceed with the disassembly.

   b. You may type * <LABEL> which will disassemble the BOOT sector of the diskette in the drive specified, where <LABEL> is the name you want printed on the page heading of the disassembly listing. Note: the brackets are not typed and are used only for purposes of explanation.

   c. You may type # <LABEL> to disassemble a program from the Atari's memory. If you answer with this prompt you will be asked for the Starting Location and then the Ending Location. Each response will require you to input four Hexadecimal Digits for each of the locations.

6. You may abort the disassembly at any time by pressing the <START> key on the Atari. When this happens, you will be returned to the beginning of the disassembler for another disassembly.

C.1.7 INHANC.V11

This program will NON-DESTRUCTIVELY reformat your single density, NON-COPY PROTECTED diskettes for the OPTIMUM INTERLEAVE FACTOR FOR READ OPERATIONS. THE LINE PRINTER ENABLE SWITCH SW1-6 MUST BE UP TO MAKE THIS PROGRAM OPERATIONAL. After the Atari has transferred the code in the DATA statements to the MMF disk drive, it will read an entire track into the PCP-80's internal RAM memory, reformat the disk using the optimum sector skew for read, and rewrite the information back onto the track. It requires no operator intervention other than to put the disk that you wish to ENHANCE into the disk drive. The program is set up initially to work with D2:. If you have a single drive, you must change line 1070 as noted for the other programs.

©1983, Micro Mainframe

C.1.8 STFORMAT.V11

This program gives you the ability to custom format any track on a diskette. As an alternative, you may use MICROZAP if your only desire is to alter the sector skew. You will find that MICROZAP is much easier to use for this purpose. STFORMAT.V11 is primarily used to change the track layout to suit your own preference. The discussion below is necessarily brief and we refer you to the Western Digital literature for the WD1771 and WD179X disk controllers for additional information. The spaces are inserted in the DATA statements below to enhance their readability and must not be present in the actual program as we have mentioned previously.

1070 DATA (32) 00 00 00 (02) (02) .....

The first byte circled in line 1070 (32 in this case) tells the program to use drive D2:. If, instead, you wish to use drive D1: change the 32 to a 31.

The second byte circled tells the format program which PHYSICAL TRACK NUMBER you wish to format, while the third byte represents the track number you wish to write on the track. As written, line 1070 tells the firmware in the MMF disk drive to format physical track 2 on drive D2: as logical track 2.

You may alter the table defined in lines 1090 & 1100. It contains the track layout for a single disk track. The first 22 bytes are the Z-80 code that performs the job. The actual track layout table begins with byte 23 of line 1090 which is a 14 Hex.

1090 DATA . . . . . 14 FF 06 00 0F FF 06 00 00 0B

1100 DATA FF 06 00 80 E5 0A FF FF 09 12 07 10 05 0E 03 0C 01
0A 08 11 06 0F 04 0D 02 0B 00

We have expanded the definitions of this table for you below. Note that most of the bytes are used in pairs. We have supplied both the Hex numbers and their Decimal equivalents.

HEX	DECIMAL	BYTE	FUNCTION
14	20	FF	GAP 1
06	06	00	SYNC BYTES
0F	15	FF	SPEED GAPS
06	06	00	SYNC BYTES.
0F	15	FF	SPEED GAPS
06	06	00	SYNC BYTES
00	00	00	SIDE SELECT
0B	11	FF	SPEED GAPS
06	06	00	SYNC BYTES
80	128	E5	DATA FIELD
0A	10	FF	GAP 3
FF	255	FF	END GAP

The remaining bytes in line 1100 define the physical sector skew with which we wish to format the track. This table is defined below. The values presented are for single density, optimum read speed which is identical to the results obtained when using INHANC.V11. Any change to the sector skew may speed up or slow down the read or write speed. The process defined is repeated for each sector on the track. Diskettes that are INHANCED will probably read much slower on drives by other manufacturers, since the MMF drive is one of the fastest drives available.

 ©1983, Micro Mainframe

If you experiment with the skew, you will find that certain operations will be faster with the sector skew given and some will be slower. Although it may be possible to increase the read speed of a disk, you may discover a corresponding degradation in the write speed to the disk.

SECTOR LAYOUT TABLE

PSECT	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	16	17	18
HEX	09	12	07	10	05	0E	03	0C	01	0A	08	11	06	0F	04	0D	02	0B
DECIMAL	09	18	07	16	05	14	03	12	01	10	08	17	06	15	04	13	02	11

The table above shows the correlation between physical sectors and logical sectors. PSECT is the physical sector position. The logical sector position is given in both HEX and DECIMAL notation.

C.1.7 SPEED.V11

This program will dramatically increase the speed of loading your programs into the Atari, as it enables the disk drive to read an entire track at once instead of just a sector. Note that the MMF disk drive is through reading long before Atari can take the information. SPEED.V11 DOES HAVE ONE DRAWBACK, HOWEVER, FOR WHICH THERE IS NO IMMEDIATE SOLUTION. IF YOU CHANGE THE DISKETTE FROM WHICH YOU ARE READING, YOU MUST PRESS THE SYSTEM RESET BUTTON ON THE ATARI BEFORE YOU CAN READ A DIFFERENT DISKETTE.

This concludes the description of the programs on your utility diskette. Micro Mainframe may release additional material from time to time which you may obtain for a nominal copying charge of $15 plus $2 shipping and handling. **Please include a check for $17 when you send your diskette in. You must have your original diskette to be eligible to receive these additional programs. New programs will be issued only to holders of original serialized Micro Mainframe diskettes.**

MICRO MAINFRAME
11325-A SUNRISE GOLD CIRCLE
RANCHO CORDOVA, CA 95670
(916) 635-3997

 ©1983, Micro Mainframe

MICROZAP®

DISK ZAP UTILITY PROGRAM

MICROZAP® is Micro Mainframe's utility program that allows you to examine and modify your diskettes. We give this program away FREE with every drive purchased along with several other useful utilities. Some of the features of MICROZAP® will work only with the Micro Mainframe MF-XX8X series disk drives and these are annotated in the Main Menu. Most other manufacturers of Atari Peripherals charge you exorbitant sums of money for programs that are not nearly so useful as MICROZAP® and DISASSEM. Providing programs of this caliber at NO CHARGE is Micro Mainframe's way of saying THANK YOU FOR PURCHASING ONE OF OUR PRODUCTS. Micro Mainframe is light years ahead of others when it comes to product capability, utility, and service.

The following is a list of some of the things you may do with MICROZAP®: EXAMINE, ALTER, VERIFY, AND COPY DISK SECTORS; CALCULATE PHYSICAL TRACK and SECTOR NUMBER given the Atari sector number; FORMAT the ENTIRE DISKETTE; FORMAT a STANDARD TRACK; SET DATA MARKER to F8 instead of FB; and FORMAT A TRACK WITH ATTRIBUTES.

The FORMAT TRACK with ATTRIBUTES feature is, by far, the most versatile utility ever written for the Atari computer. It allows the user to custom format a track with CRC errors, F8 data markers, duplicate sectors, and missing sectors; You may also specify any sector skew that you might desire.

You may use the EXAMINE command to look at disk sector information both in HEXADECIMAL and ATASCII. If you find information contained within a disk sector which you wish to alter, you may modify it using the MODIFY mode. MICROZAP® allows you to verify any number of disk sectors from a single sector to the entire diskette. The COPY command allows you to move any sector or group of sectors from one diskette to another or from one place on a diskette to another. The FORMAT DISK and FORMAT DISK TRACK command allows you to format either the entire disk or a single track on the diskette. When using the single track format you may optionally specify which sector header track number is to be formated on any specific physical track.

The main menu lists the following nine options:

1. DISPLAY DISK SECTOR
2. VERIFY DISK SECTOR(S)
3. COPY DISK SECTORS
4. CONFIGURE ATARI FOR DRIVE TYPE(S)
5. FORMAT DISK
6. CALCULATE TRACK, SECTOR
7. FORMAT TRACK (MMF DRIVE V11 ONLY)
8. WRITE F8 DAM (MMF DRIVE V11 ONLY)
9. ATB TRACK FORMAT(MMF DRIVE V11 ONLY)

Functions are selected by pressing the the number of the action desired. It is not necessary to press the <RETURN> key to select a function. After you have made your selection, you may be prompted by the program to input additional information. Some of these answers will not require a <RETURN> while others will. Please note that all information requested by MICROZAP® requires that the values be in Hexidecimal (Hex). If you are not familiar with this number system, you must become familiar with it prior to using MICROZAP®. We cannot assist in this procedure, as there are several magazines and books that explain this useful number system. We will explain each of the individual commands available in the sections numbered below corresponding to the MENU number.

 ©1983, Micro Mainframe

1. DISPLAY DISK SECTOR

Depressing the <1> key will get you into the display disk sector mode. The computer will ask for the DRIVE number. Press any drive number desired from 1-8 on which you wish to examine a disk sector. It is not necesary to press <RETURN> after the drive has been designated. The drive must exist physically in the system and be configured in order for this feature to work.

Once the drive number key is pressed the computer will immediately ask for the SECTOR number. The program is now requesting you to input HEXIDECIMAL value corresponding to the sector you desire to examine. This value must be within the allowable range of sectors for the disk drive you are using (this is normally between 1 Hex and 2D0 Hex for a 40 track disk drive). Once you have entered the number depress the <RETURN> key. The computer will respond by reading the disk sector and displaying the sector data in both Hex and ATASCII. The COMMAND STATUS and FDC STATUS are displayed also to aid you in detecting CRC errors and other errors found during sector read operations.

The COMMAND and FDC STATUS displays are valid only under the following conditions: when the FIRST read from the menu is performed; or when a disk error, serial error, or command frame error OTHER than a data marker error is found. You may optionally force a status request by depressing the <CONTROL> key and the <S> key simultaneously (<CONTROL> S). This will cause the Atari to issue a request status command and update both the command status and FDC status.

From the sector display you may scroll the sector data up and down by using the <CONTROL> UP ARROW and <CONTROL> DOWN ARROW; you may move to the END of the sector using the <CONTROL> E, or the BEGINNING of the sector using the <CONTROL> B key. You may enter the modify mode as described below.

You may advance to the next LOGICAL sector by depressing the <+> key, move back a sector by pressing the <-> key, or advance to the next track (18 sectors forward) by depressing the <SHIFT> key and the <+> key simultaneously. You may move back a full track (18 sectors backwards) by depressing the <SHIFT> and the <-> key.

You may return to the Main Menu by depressing the <X> key.

To enter the MODIFY mode depress the <M> key from the DISPLAY mode. The computer will respond by placing the see-through cursor on the first byte of the sector data. You can move the cursor over any byte contained within the sector by using the normal control arrows.

To alter the data under the cursor, simply enter the appropriate Hex value in the location you wish to change. To abort the changes press <CONTROL> Q. To reread the sector ignoring any changes depress <CONTROL> R. Using either of these key combinations will reread the sector and place you back in the normal sector display mode.

To make the changes permanent on the diskette, press the <RETURN> key. The diskette must not be write protected when you do this. The updated information is written to the disk and then read back leaving you in the normal sector display mode.

2. VERIFY DISK SECTOR(S)

Enter the VERIFY mode by pressing the <2> key from the Main Menu. The computer will prompt you for the SOURCE DRIVE number. Depress the number that corresponds to the drive number that contains the diskete upon which you wish to verify sectors.

The computer will prompt you for the SOURCE START SECTOR number. Type a Hex number in the range appropriate for your disk drive to identify the group of sectors you wish to verify. Press the <RETURN> key when you have completed this.

The computer will prompt you for the NUMBER OF SECTORS. nter E a Hex number in the range appropriate for your disk drive and for the number of sectors you wish to verify, press the <RETURN> key when complete.

©1983, Micro Mainframe

The computer will display the following line of information:

XXXX :ERROR # 0000, TOTAL ERRORS XXXX

Where:

XXXX--> TOTAL ERRORS THIS VERIFICATION.

XXXX----------> LAST SECTOR NUMBER FOUND TO HAVE AN ERROR.

XXXX==========> CURRENT SECTOR BEING VERIFIED.

Once the verification is complete the message START will appear in the current sector number area. Depress the START key to return to Main Menu.

3. COPY DISK SECTORS

COPY DISK SECTORS will allow you to copy disk sectors from one diskette to either another diskette or to a different area on the disk; it will allow you to move as many sectors as you wish to move.

The computer will prompt you for the SOURCE DRIVE number, which may be any drive on the system. The computer then will prompt you for the SOURCE START SECTOR number. Please input a Hex number that is within the range appropriate for your diskette and drive combination.

The computer will prompt you for the DESTINATION DRIVE number, which may be any drive on the system. The computer then will prompt you for the DESTINATION START SECTOR number. Your response should be a Hex number within the range of your diskette and drive combination.

The computer will prompt you for the NUMBER OF SECTORS ?. Type in the Hex number of sectors you wish to copy.

If you specify the same number for both SOURCE and DESTINATION drives, the computer will default to the single drive copy mode and will prompt the user to insert the appropriate diskette before action is taken. If you have a multiple drive configuration, you may specify different drive numbers for each drive. The computer will proceed to copy the sector information from one disk to the other.

4. CONFIGURE ATARI FOR DRIVE TYPE(S)

This command tells the Atari to configure for the type of drives connected (which will be either 810 single density, or 815 double density types). This feature allows you to configure the Micro Mainframe for a density that is opposite that upon which you booted up.

5. FORMAT DISK

This formats the entire disk using the standard Atari format. The computer will prompt you for the DESTINATION DRIVE number. Type the drive number of the drive you wish to format. This number may specify any drive that is connected onto your system.

6. CALCULATE TRACK, SECTOR

This command will convert an Atari sector number to the Physical Track, and Sector Number. The computer will ask CALCULATE WHICH SECTOR NUMBER and requires that you input a Hex value between 0-nnnn. NOTE: **single byte track numbers will roll over.**

7. FORMAT TRACK (MMF DRIVE V11 ONLY)

Format track is useful in many ways but may only be used on the Micro Mainframe drive. The command will allow you to format a single track on a diskette and use any header information you wish for the track number. **Please Note: Neither COMMAND 7 nor COMMAND 9 will format a**

track unless it locates at least one sector on the specified track. We have done this purposely so that if the disk head becomes lost it will not write over information that you do not desire to overwrite. This means that if you have previously burned a track, you must reformat using COMMAND 5 or the format command from DOS.

After you have issued COMMAND 5 the computer will prompt you for the DESTINATION DRIVE number. Type the number of the disk drive containing the diskette upon which you wish to format a track. The computer then will query PHYSICAL TRACK NUMBER TO FORMAT? Type a Hex number within the range of tracks on your drive and diskette. The computer finally will prompt you TRACK HEADER # (RETURN = PHYSICAL). The value requested is the track number you desire to be placed into the header information for that specific track when it is formatted on the disk. If you respond with <RETURN> the program will use the same number you specified for the physical track. This is the normal action to take if you desire to read the track.

If you desire to burn the track entirely, you may respond with a track number other than the physical number of the track - eg. if you are on track 5, you can tell the controller that you wish to write the header specifying that it is track 7. When the controller is asked to read track 5, the drive will step out to physical track 5 and find the header for track 7. The result is that the controller will return a SEEK ERROR thereby giving the appearance of a burned track.

8. WRITE F8 DAM (MMF DRIVE V11 ONLY)

This command allows you to alter the data marker written as the header of the sector to an F8 (DELETED DATA MARKER). The sector information is not changed. This feature changes only the data marker for that sector. If you wish to change a data marker from an F8 Hex to an FB Hex, use the display sector command and enter the MODIFY mode and without altering any information press the <RETURN> key. This will cause the controller to write the sector back to disk with the normal data marker (FB) Hex.

The computer will prompt you for the DESTINATION DRIVE number. Type the number of the drive containing the diskette you wish to change. The computer will prompt TRACK NUMBER FOR DAM BYTE CHANGE? This is the physical track where the sector resides that you wish to change. The computer will then ask SECTOR NUMBER FOR DAM BYTE CHANGE This is the sector in which you wish to change the DAM byte to an F8.

9. ATB TRACK FORMAT (MMF DRIVE V11 ONLY)

This command allows you to format a track and specify the attributes. You may use it instead of COMMAND # 7 to format a single track along with the same restrictions as noted above. If you wish to format the track in the standard format, just press <RETURN> when the Attribute Table is displayed and it will write the same type of track as COMMAND # 7. Further instructions in the use of this program are outlined below.

After you have specified COMMAND 9, the computer will prompt you for the DESTINATION DRIVE number. This is the drive which contains diskette upon which you wish to place the special format.

The computer will prompt you for the PHYSICAL TRACK NUMBER TO FORMAT. The value requested here is the physical track that you wish to position the read/write head over for the format.

The computer will prompt TRACK HEADER # (RETURN = PHYSICAL). This value is the number that will be placed into each sector header during the format. You may specify a value that is different from that of the PHYSICAL number if you wish to BURN a track as noted in COMMAND 7 above.

The computer will then display the current attribute table. Although this table may seem confusing at first, it offers tremendous flexibility for formatting unusual formats as you will soon discover.

©1983, Micro Mainframe

ATTRIBUTE TABLE

```
SPECIAL TRACK FORMAT, ID 31, TRACK 27
POS:SEC:ATB:POS:SEC:ATB
  1: 09: 01:  2: 12: 01
  3: 07: 01:  4: 10: 01
  5: 05: 01:  6: 0E: 01
  7: 03: 01:  8: 0C: 01
  9: 01: 01: 10: 0A: 01
 11: 08: 01: 12: 11: 01
 13: 06: 01: 14: 0F: 01
 15: 04: 01: 16: 0D: 01
 17: 02: 01: 18: 0B: 01

ATB : 00=NO SECTOR, 01=NORMAL SECTOR
ATB : 02=F8 DAM   , 03=CRC ERROR

PRESS <RETURN> TO WRITE TRACK
PRESS <CONTROL> <Q> TO ABORT
```

The cursor will be placed upon position 1, which corresponds to a sector number of 9. You may move the cursor to another cell by holding the <CONTROL> key and one of the arrow keys (Left Arrow, Right Arrow, Up Arrow, or Down Arrow). To alter any byte in the Attribute Table, enter a Hex number that corresponds to the change you wish to make.

Note that there are 18 sectors and 18 positions. Each position accounts for one sector position on the track and its physical location in relation the track. Therefore position 1 is the first position on the track and position 18 (12H) will be the last sector on the track. It does not matter what sector number it is, we are talking only about physical location on the track.

You are given the capability to change the sector interleave (skew) by specifing each sector number in each of the 18 postions. If you specify a sector skew different from the default value in the Attribute Table, it will generally slow down Disk I/O substantially, as the default value given is the optimum for read operations. It will not necessarily be the optimum for write operations. You can also specify that duplicate sectors be written on a given track by entering the same sector number more than once or as you wish; however, you will loose one sector position for each duplicated sector number you have specified, as there are only 18 possible sector positions on any given track.

You can further specify the type of sector you wish to write by DAM byte type: eg. - 01 ATB writes an FB (Data Marker) and an 02 writes a F8 (Deleted Data Marker). Specifying 03 will cause the FDC to write a sector with a bad CRC value.

Once you have edited the ATB table to reflect the way the track is to be written, depressing the <RETURN> key will program the MMF drive regarding your selections and it will write that track to the diskette. Please note that the Line Printer Switch (SW1-6) must be enabled to perform any of the special options available only with the Micro Mainframe disk drives. COMMANDS 7, 8, AND 9 WORK ONLY WITH THE MICRO MAINFRAME DISK DRIVE.

SAMPLE PROCEDURE TO DUPLICATE A DISKETTE

This procedure will allow you to make backup copies of certain diskettes that you are unable to copy by any other means. It is not automatic, and requires some effort on your part.

1. Step through the diskette that you wish to duplicate using the DISPLAY SECTOR(S) feature of MICROZAP®. Write down the numbers of any sectors not found and the status of the FDC chip at that time.

2. Format a BLANK diskette by using COMMAND 5 of MICROZAP®.

3. Using COMMAND 9 of MICROZAP®, change the sector information for those tracks you have found to contain missing sectors, CRC ERRORS, or sector timing errors. Hit <RETURN> to write these changes to the diskette. PLEASE NOTE: **If you have burned a track on your diskette you will have to reformat the disk with COMMAND 5, as MICROZAP® will not write the updated information to the target diskette that has missing track numbers. It will write only to a newly formatted track that has no special formatting.**

4. Repeat the above for any additional tracks and sectors that are either missing or have unusual situations.

5. Using <SECTOR COPY>, copy the sector contents for each sector of the program appearing on your master diskette. You may do this either singly or multiply if you have a range that is consecutive.

SOME HELPFUL HINTS

1. You may burn an entire track by specifing a TRACK HEADER that is different from the PHYSICAL track number as noted above. You may optionally use COMMAND 9 and replace the default value of 01H in the ATB column with 00H for each of the sector headers.

2. You will need to obtain a copy of the WESTERN DIGITAL data sheets and applications notes for the WD-179X and WD-1771 floppy disk controllers. This will explain the bit pattern found in FDC STATUS register that is displayed when you read a sector. You may obtain further information from Western Digital Corporation at the following address:

WESTERN DIGITAL CORP.
2445 McCabe Way
Irvine, CA 92714
(714) 851-1221

For further information pertaining to the COMMAND STATUS as displayed, we refer you to the **Atari Technical Reference Manual** which is obtainable through most Atari dealers.

Micro Mainframe cannot provide any information relating either to Copy Protection schemes or how to reproduce any Copy Protected programs. We have provided you with the tools necessary to determine what is being done to protect some of these programs so that you may back up your legitimate copies. Due to the highly transient nature of copy protection schemes, it is next to impossible to write an automatic copy program that will work every time with the new copy protection schemes coming out almost weekly. While the manual method is more complex to follow, it should work with almost any conceivable scheme.

MICROZAP®, when used in combination with PRTTRK, PRTSEC, DISASSEM, and the READ TRACK Command built into your MF-XX8X should give you all of the tools necessary to back up protected programs. You must obtain the Western Digital Manuals previously mentioned and the Atari Technical Reference Manual in order to completely understand disk procedures as related to your Atari. Good Luck in your efforts.

©1983, Micro Mainframe

MF-168X SERIES

DISK DRIVE FOR ATARI

BY MICRO MAINFRAME

Congratulations!! You have just reached a new dimension in computing power by the addition of a Micro Mainframe disk drive to your Atari computer system. We think that you will find the difference in performance between the Micro Mainframe disk drive and other disk drives for the Atari is like comparing the performance of a Boeing 707 to a Lear Jet.

This manual is valid for all Micro Mainframe disk drive Models MF-XX8X. The nomenclature for the MMF Atari Disk drive is as follows:

where:

XX = 16 for a 16K unit; 64 for a 64K unit.
8 designates that the processor is a Z80.
X is 1 for a single drive unit and 2 for a dual drive unit.

Therefore, an MF-1681 designates a 16K single drive unit; an MF-6481 designates a 64K single drive unit, etc.

We have divided this manual into sections to provide several levels of knowledge. The Operations Manual (Section I) will provide you with the basic information necessary to operate your new disk drive with a minimum expenditure of time and effort. Some of you may never desire to know any more about your disk drive than the information provided in the Operations Manual.

Section I covers six important areas; (1) Using the drive as shipped; (2) Booting the operating system diskette; (3) Configuration of the line printer switch; (4) Making a Double Density diskette; (5) Diagrams and Illustrations; and (6) When Things Go Wrong.

Section II is the part of the Technical Manual that describes the Port definitions for the Z-80 computer and the configuration procedures for the DIP switches SW-1 & SW-2. Section III is the continuation of the Technical Manual and describes the command structure which may be sent to the PCP-80 via the line printer port. Appendix A is an explanation of the differences between the single and double density controller chips used by Atari and Micro Mainframe respectively. Appendix B briefly describes a feature we have implemented for future operating systems to double the 19.2 KB transfer rate to 38.4 KB.

In order that you might begin immediately, we recommend that you first try the disk drive in its original configuration before you attempt to make any changes. THREE OPERATING SYSTEMS ARE CURRENTLY SUPPORTED BY THE MICRO MAINFRAME MF-XX8X SERIES DISK DRIVE. THESE ARE: ATARI DOS 2.0S, OS A+ BY OPTIMIZED SYSTEMS SOFTWARE (VERSIONS 4.0 & 4.1), AND MYDOS BY WORDMARK SYSTEMS. YOU MUST PURCHASE ONE OF THESE OPERATING SYSTEMS FROM YOUR DEALER IN ORDER TO MAKE YOUR MICRO MAINFRAME DISK DRIVE WORK!

NOTE: THE WARRANTY CARD SUPPLIED WITH YOUR MF-XX8X SERIES DISK DRIVE MAY STATE THAT THE WARRANTY PERIOD IS 180 DAYS. THIS IS DUE TO AN ERROR BY OUR PRINTER AND HAS BEEN MARKED OUT IN RED INK. WE HAVE INSERTED THE CORRECT WARRANTY PERIOD OF 90 DAYS BY HAND WHICH IS THE CORRECT WARRANTY PERIOD. WE APOLOGIZE FOR THE INCONVENIENCE.

 <c> 1983, Micro Mainframe

MF-168X SERIES

DISK DRIVE FOR ATARI

BY MICRO MAINFRAME

SECTION I - OPERATIONS MANUAL

1.0 BRINGING UP THE SYSTEM FOR THE FIRST TIME

1.1 After passing an exhaustive checkout procedure, your MMF drive has been factory programmed to emulate a standard Atari disk drive prior to shipping: **ie - Single Density, defined as Drive 1, No Line Printer operation, and the Automatic Density Recognition mode is ON.**

1.2 Carefully unpack your MMF disk drive from the shipping carton. **Do not discard the shipping carton, as you must use it to return the disk drive to MMF in the unlikely event that your unit needs repairs.** MMF WILL NOT ACCEPT DRIVES FOR REPAIR UNLESS THEY ARE RETURNED IN THE ORIGINAL SHIPPING CARTON. The following procedure assumes that you have a Micro Mainframe disk drive attached to your Atari computer as the only disk drive. If you will be using the MMF drive in conjunction with drives provided by other manufacturers, we still recommend that you initially connect the MMF drive as if it were the only drive on the system to check out the drive operation with your particular computer. Once you have determined that the MMF drive is functioning properly with your system, you may then skip to Sections 2 & 3 for some sample configuration procedures using the advanced features of the Micro Mainframe MF-XX8X series disk drives.

1.3 Referring to FIGURE 5-1, locate the cable marked TO ATARI COMPUTER on your MF-168X disk drive and connect it to the expansion socket on the right side of your Atari computer. This is the only connection you are required to make to your Atari computer system.

a. Connect the AC POWER CABLE to a suitable source of AC power (115 VAC).

b. Turn on the Drive Power Switch (Designated S1).

2.0 BOOTING THE THE OPERATING SYSTEM DISKETTE

2.1 You may boot up with or without the Basic Cartridge installed in your Atari computer (Left socket on the 800 computer system).

2.2 First, locate the diskette containing the operating system in the package you obtained either from ATARI, Optimized Systems Software, or Wordmark Systems. Referring to FIGURE 5-2 for the correct positioning of the diskette, open the door on the disk drive by turning the handle on the front of the disk drive to the HORIZONTAL position. Insert the diskette as shown, and close the door on the disk drive by turning the handle to the vertical position. Make certain the diskette is pushed in as far as it will go prior to closing the door!

2.3 You are now ready to tell the disk drive to BOOT the operating system diskette. BOOT literally means BOOTstrap, and as used here means to load a special program from the master diskette (the one you purchased from Atari, OSS or Wordmark Systems). The boot software initializes certain parameters in your Atari computer, and then loads the operating system into the memory of your Atari. We tell the Atari to BOOT the disk drive by performing one of the two operations described below. YOU MUST FIRST CONNECT THE DISK DRIVE TO THE ATARI AND TURN ON THE DISK DRIVE POWER SWITCH BEFORE YOU ATTEMPT TO BOOT THE DOS.

a. Open and then close the cartridge slot on the Atari 400 or 800 as if to install a cartridge.

 <c> 1983, Micro Mainframe

b. Turn the Atari power switch off and then back on (This switch is located on the right side of the Atari 400 or 800 computer system).

Every time that you open and close the cartridge door or turn the power switch on the Atari off and back on, the disk drive will attempt to BOOT the diskette you have inserted into the disk drive, PROVIDED THAT THE DISK DRIVE IS CONNECTED AND THE POWER IS TURNED ON.

c. Insert the Master DOS diskette at this time (don't forget to close the door on the disk drive), turn up the volume on your TV set and give the command to BOOT by one of the two methods outlined above. You should hear a beeping sound emanating from the TV speaker indicating that the Atari is loading the operating system from your disk drive. Upon completion of the load you will get one of two messages on the screen.

1. If your Basic Cartridge IS installed, the computer will load the Operating System and return to Basic and display READY. You should type DOS and hit the <RETURN> key to get into the operating system. You will be given a Menu of operations that you can perform at this time. Please refer to the Atari, OSS or Wordmark Systems DOS manual for further information on the use and features of the operating system.

2. If the Basic Cartridge IS NOT installed, the computer will go immediately to DOS and present the menu described in 1 above. You should now become familiar with the procedures outlined in the manual included with your operating system.

d. Your disk system should now be operational. If the system did not perform as indicated, you have probably forgotton to perform one of the steps. Recheck your work before you going into panic. If you still have problems then please refer to Paragraph 5 of the Operations Manual entitled WHEN THINGS GO WRONG!

If you are not familiar with disk operating procedures, we refer you to the DOS manual that came with your operating system. We recommend that you make several back up copies of your operating system at this time, and put your master diskette in a safe place. As this manual is not intended to replace your DOS manual, you should now become familiar with all of the procedures for your particular operating system. Once you understand how the DOS operates, you may wish to explore the other sections of this manual.

3.0 SELECTION OF THE LINE PRINTER OPTION

3.1 Configuring your MMF disk drive to support the Centronics compatible line printer is as simple as the flip of a switch. Refer to FIGURE 5-1 for the location of the configuration switches. Examine Switch SW-1 and locate switch 6 (the sixth switch from the left). As shipped from the factory, this switch is turned off. To enable the line printer option, you must turn this switch on. FIGURE 5-3 shows the proper position of this switch to enable the line printer. You must now turn off the power to the disk drive and leave it off for 10 seconds before turning it back on. **This is because the disk drive reads the condition of SW-1 and SW-2 only upon power up.** If you forget to perform this step, the printer will not operate until you turn the power off and back on as described above.

4.0 MAKING A DOUBLE DENSITY DISKETTE

Contrary to what you may have been told, Atari 2.0S DOS will run in the Double Density mode. It is very simple to accomplish if you follow these instructions.

 <c> 1983, Micro Mainframe

4.1 PROCEDURE FOR USE WITH A SINGLE DRIVE SYSTEM

Single drive users must execute the program named DOUBLE located on the program disk supplied by MMF. These diskettes will be sent out upon receipt of your completed warranty card. The program is self prompting and will not require any further documentation.

Use The SETDBL Program Supplied

4.2 PROCEDURE FOR USE WITH A DOUBLE DRIVE SYSTEM

You must remove your Basic cartridge from the computer prior to performing the following procedure.

1. Turn the power off on the MF-XX8X disk drive. This switch is located on the rear panel of the drive near the bottom. See FIGURE 5-1 for the location of the configuration and power switches.

2. Drives D1 and D2 are factory configured to be single density. Leave D1 set to single density and change D2 to double density by setting SW2-3 to the ON (UP) position. See FIGURE 5-4 for this configuration.

3. Turn on the power switch on the MF-XX8X drive system. This action will cause the PCP-80 CPU to read the settings of the configuration switches and D1 will be set for single density and D2 will be set for double density. NOTE: **The condition of the configuration switches will only be read when you have performed a power on reset or if you push the reset switch SW-3 (see** FIGURE 5-1 for the location of this switch).

4. Insert a copy of your master DOS diskette in drive D1 and close the door by turning the handle to the down position. Leave the door open on drive D2. Boot the DOS diskette by turning your Atari off and back on. This action will cause the Atari computer to configure drive D1 single density and drive D2 double density.

5. Insert a blank diskette in drive D2 and close the door.

6. Issue the I command from Atari DOS specifying D2 when asked which drive to format. The Atari will now format D2 as a double density diskette.

7. At the DOS ready prompt, issue the H command. The drive spec will be D2. This will cause the Atari to copy the system files onto the double density diskette you just formatted in drive D2. That is all there is to it. You may desire to format several diskettes as double density at this time and transfer your single density NON-PROTECTED files over to the double density diskettes. PLEASE NOTE: **you will not be able to copy any of the copy protected or self booting diskettes onto double density in this manner, as this procedure only allows programs that exist as loadable files to be copied. You are on your own if you try to move such programs over to double density, as a discussion of this topic is beyond the scope of this manual.**

 <c> 1983, Micro Mainframe

5.0 FIGURES AND ILLUSTRATIONS

6.0 WHEN THINGS GO WRONG

If you have read this far, then you are, most likely, having problems with one or more of the procedures outlined in the manual above. If this is the case DO NOT PANIC as you have probably overlooked a small, but important step in one of the procedures outlined previously. This section of the manual is designed to help you through your problem.

Since even the most basic computer system is infinitely complex when compared to other electronic systems, we must rationally approach the problem. If we do not, we may end up chasing our tails in a circular pattern and never locate the problem.

When we have a problem with the system, it is extremely important to define what is operating properly and what is not. This is the key to the successful troubleshooting of a complex system.

Do not overlook such simple factors as the following: Check to see if the door on the disk drive or drives is closed; Check all cables to see if they are properly connected; Check the diskette to make certain you have properly inserted it in the disk drive; Check the configuration switches to make certain that they are configuring the system for what you want to do; Check to see if the power is on to both the Atari and the disk drive and make certain that the disk drive has been powered up first; etc.

Consider the following example: Your disk drive works properly in the Single density mode, but you cannot make a double density diskette.

We should first determine if the computer recognizes the disk drive in the system device table. The fastest way to accomplish this is to turn off the power to both the Atari and the MF-XX8X disk drive. Check the position of the configuration switches to make certain they are configured properly while the power is off. This delay will give the power supply capacitors an opportunity to discharge. Now, turn on the power to the disk drive only. Insert a standard single density diskette into D1 and close the door. Turn on the power to the Atari and the diskette should boot. If it does not, then check such things as whether the cable from the disk drive to the Atari is properly seated in the Atari's expansion connector.

We will probably discover that we forgot to perform one of the operations in the right order and most likely, we will have fixed the problem during our checkout.

Please remember: If one disk drive is working in any mode, we have a good indication that the CPU and disk controller in your MF-XX8X are working properly. If the second and succeeding drives are not working, we could have a problem with either the drive mechanism itself, with the cable connections, the power supply for the additional drive (if a separate power supply from the one in your MF-XX8X unit), or your configuration procedure.

Before you call Micro Mainframe's customer service number, please perform the procedures outlined above. It will greatly help us to assist you, and may even save you a Toll call. Please remember that the power supply in the MF-XX8X will power only two disk drives. If you attempt to run more than two drives on this power supply, you may damage the supply. Micro Mainframe will not warranty power supplies damaged in this manner.

If you have attempted the procedures outlined above and still cannot make your disk drive function, please call the Micro Mainframe Customer Service Department at the number listed below. Please have the serial number of your drive available for the technician when he answers. PLEASE NOTE: MICRO MAINFRAME'S HOURS OF OPERATION ARE FROM 8:30 AM TO 5:00 PM PACIFIC TIME. YOU MUST CALL BETWEEN THOSE HOURS ONLY. DO NOT SEND YOUR DRIVE BACK TO MICRO MAINFRAME WITHOUT HAVING FIRST OBTAINED AN RMA NUMBER FROM THE SERVICE TECHNICIAN. MICRO MAINFRAME WILL REFUSE ALL DRIVES SENT BACK WITHOUT THE RMA NUMBER CLEARLY MARKED ON THE OUTSIDE OF THE BOX.

11325 SUNRISE GOLD CIRCLE
BUILDING A
RANCHO CORDOVA, CA 95670
(916) 635-3997

 <c> 1983, Micro Mainframe

MF-XX8X DISK DRIVES

BY MICRO MAINFRAME

SECTION 2 - TECHNICAL MANUAL

PORT DESCRIPTION OF THE PCP-80 AND FPC-80

The MF-XX8X series disk drive system for the Atari computer uses a powerful Z-80A CPU to send, receive, and assemble the information going to and from the Atari computer system. The Z-80 Computer architecture is considerably different from the 6502 CPU used in the Atari. The Z-80 has more internal registers than than the 6502 and the I/O may be either port mapped or memory mapped. For those of you who may be interested in learning more about the Z-80 CPU, we refer you to the Z-80 Technical Reference Manual (available from Zilog or Mostek), as a discussion of these topics is beyond the scope of this manual.

Ports will be described in ascending numerical order. As used in this discussion, PCP-80 refers to the Z-80A CPU Card, FPC refers to the Floppy/Printer Controller, and Host Adapter refers either to the Z-80 Hot Clip/HA-3A combination or to the STD-HA-1. All products listed are designations of Micro Mainframe individual component parts.

HARD DISK HOST ADAPTER

PORT 0C0H INPUT/OUTPUT

BIT	DESCRIPTION
ALL	Bi-Directional Data/Command, Completion Status

PORT 0C1H OUTPUT DEFINITIONS

BIT	DESCRIPTION
7	Output Bit Software Reset For Hard Disk Controller
6	Output Bit Select Hard Disk Controler
3	Output Bit Queue/Dequeue
0	Output Bit Controller ID

Queue mode is used when bits 6, 3, & 0 are all set during select. Dequeue is performed when bits 7, & 3 are set and output to Port (0C1H). PLEASE NOTE: **Queue & Dequeue are available only when used in conjunction with the Micro Mainframe DMX-1A Four Port Multiplexer.**

 <c> 1982, Micro Mainframe

PORT 0C1H INPUT DEFINITIONS

BIT	DESCRIPTION
7	Not Used
6	Not Used
5	Not Used
4	I/O Direction Line; 0=Host to Controller; 1=Controller to Host
3	Busy Bit; 0=Not Busy, Not Selected; 1= Busy, Selected
2	MSG
1	Command/Data; 0=Data Bytes 1=Command Bytes/Completion Status Bytes
0	Controller Data Request; 0=No Requests; 1=Requests

This concludes the Hard Disk Host Adapter Port Definitions. All remaining ports apply either to the PCP-80 CPU Module or to the FPC-80 Floppy/Printer Controller Module. We will make no attempt to separate them in this section of the manual.

PORT 0E0H OUTPUT DEFINITIONS

BIT	DESCRIPTION
7	Enable All RAM (Disable ROM)
0	Interrupt Line to Atari (1=Interrupt)

PORT 0E4H OUTPUT DEFINITIONS

BIT	DESCRIPTION
7	Enable FDC Interrupts (1=Enable)
6	Enable Drive Timeout Interrupt (1=Enable)
5	Not Used
4	Drive Size Select (1=8" Drive; 0=5 1/4" Drive)
3	Not Used
2	Not Used
1	Drive Bank Select (0=Internal; 1=External)
0	Index Pulse Enable (0=Enable; 1=Disable)

PORT 0E4H INPUT DEFINITIONS

BIT	DESCRIPTION
7	FDCINTT occurred (1); Did not occur (0)
6	Drive Timeout Occurred (0); Did not Occur (1)

PORT 0ECH OUTPUT DEFINITIONS (OPTIONAL LED DISPLAY)

BIT	DESCRIPTION
ALL	Each Bit Corresponds to Each LED indicator

PORT 0F0H OUTPUT DEFINITIONS (FDC COMMAND REGISTER)

BIT	DESCRIPTION
ALL	Write Command to Floppy Disk Controller Chip

 <c> 1982, Micro Mainframe

PORT 0F0H INPUT DEFINITIONS (FDC STATUS REGISTER)

BIT	DESCRIPTION
ALL	Read Floppy Disk Controller Status

PORT 0F1H OUTPUT (FDC TRACK REGISTER)

BIT	DESCRIPTION
ALL	Write Track Number to Floppy Disk Controller

PORT 0F1H INPUT (FDC TRACK REGISTER)

BIT	DESCRIPTION
ALL	Read Track Number from Floppy Disk Controller

PORT 0F2H OUTPUT (FDC SECTOR REGISTER)

BIT	DESCRIPTION
ALL	Write Sector Number to the Floppy Disk Controller

PORT 0F2H INPUT (FDC SECTOR REGISTER)

BIT	DESCRIPTION
ALL	Read Sector Number from Floppy Disk Controller

PORT 0F3H OUTPUT (FDC DATA REGISTER)

BIT	DESCRIPTION
ALL	Send Data to the Floppy Disk Controller

PORT 0F3H INPUT (FDC DATA REGISTER)

BIT	DESCRIPTION
ALL	Read Data from the Floppy Disk Controller

PORT 0F4H OUTPUT (DRIVE SELECT PORT)

BIT	DESCRIPTION
7	Density 0=Single; 1=Double
6	Wait on DRQ 0=Disable Wait; 1=Enable Wait on DRQ
5	Write Precomp 0=Disable Write Precomp; 1=Enable Write Precomp
4	Disk Side Number; 0=SIDE A; 1=SIDE B
3	Drive Select 3 (1) on Bank Bit 1 (PORT 0E4H)
2	Drive Select 2 (1) on Bank Bit 1 (PORT 0E4H)
1	Drive Select 1 (1)

PORT 0F4H INPUT (DIP SWITCH 2 – SEE ATARI ROM DEFINITION)

BIT	DESCRIPTION
ALL	Each Bit Corresponds to Each Switch

PORT 0F8H OUTPUT (LINE PRINTER 8 BIT DATA OUTPUT LATCH)

BIT	DESCRIPTION
ALL	Write a Byte to the Line Printer

PORT 0F8H INPUT (LINE PRINTER 4 BIT STATUS PORT)

BIT	DESCRIPTION
7	Printer Busy (1=Busy; 0=Not Busy)
6	Paper Out (1=Out; 0=Not Out)
5	Unit Select (1=Selected; 0=Not Selected)
4	Printer Fault (1=No Fault; 0=Fault Condition)
3	Not Used
2	Not Used
1	Not Used
0	Not Used

PORT 0FFH OUTPUT (SERIAL DATA LINE TO ATARI)

BIT	DESCRIPTION
0	Serial Data Bit Out to Atari

PORT 0FFH INPUT (SERIAL DATA PORT AND CLOCK PORT)

BIT	DESCRIPTION
7	Serial Data Bit in from Atari
6	Serial Command Line from Atari
5	Not Used
4	Not Used
3	Not Used
2	Not Used
1	PCP-80 19.2K Baud Clock (used for PCP-80 Serial Write Clock)
0	Atari Clock In (Used for Serial Read Logic)

ROM SWITCH SW1 DEFINITIONS NON-ATARI MODE

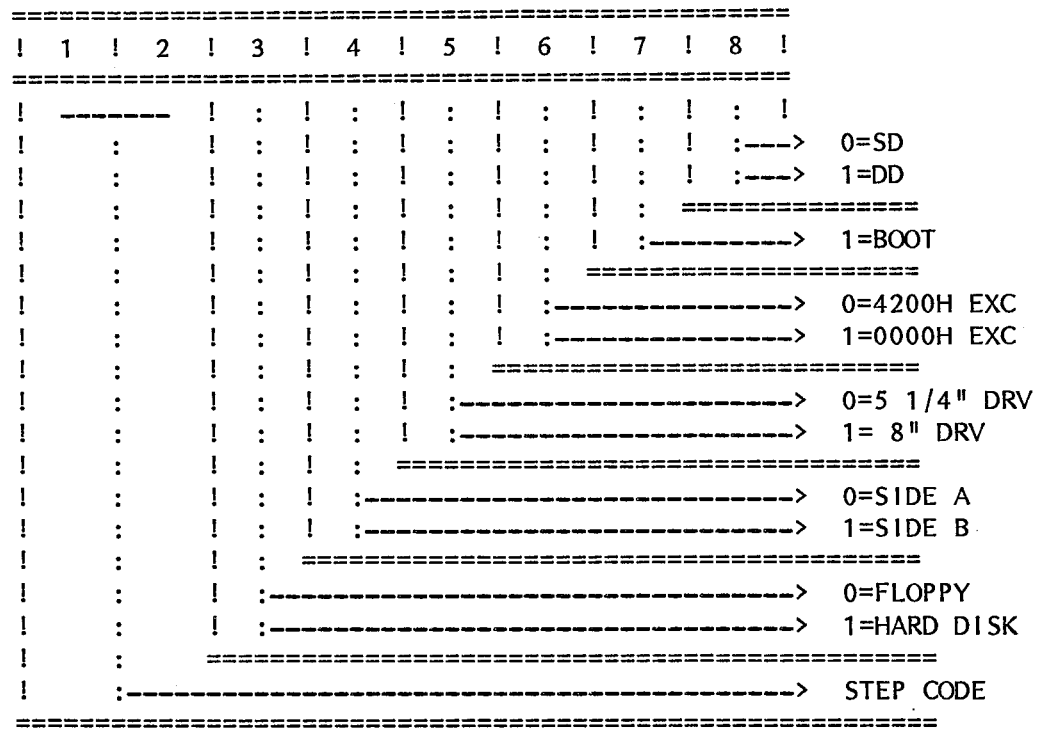
Two modes of operation are available when using the MF-XX8X series disk drives. These are the Atari Mode and the non-Atari mode. The Atari mode is the primary mode of operation and merely designates that the drive will emulate the Atari-810 disk drive with the MMF enhancements.

When specified, the non-Atari mode tells the drive to read the first sector on track 0 and load it into the PCP-80 RAM at the location specified by the position of SW1-6 (This will be either 0000H or 4200H). After this sector has been loaded program control will be transferred to the code located at that location. This feature allows the user to load operating systems such as CPM, OASIS, TRSDOS, or any other code and transfer control to the program. NOTE: **use of any of these modes will require additional hardware/software which is not included with the basic disk drive.**

Switch SW1 is used to select certain parameters necessary to operate your system. Switch SW1-7 must be ON (1) when used in the Non-Atari Mode. When Switch SW1-7 is ON as noted above, Switch SW-1 should be configured for your particular installation according to the settings below. Any settings of SW2 are ignored when SW1-7 is ON (1).

NON-ATARI MODE SW1-7 ON (1)

SWITCH SW1 CONFIGURATION



Where SD = Single Density
DD = Double Density
EXC= Execution Address
DRV= Drive

STEP CODE: The least significant two bits specify the step rate designators to be issued to the floppy disk controller during head movement operations. The setting of these switches will determine the speed at which the drives step from one track to another. Refer to the Western Digital specification manual for step rate information.

ROM SWITCH SW-1 & SW-2 DEFINITIONS - ATARI MODE

Switches SW-1 and SW-2 are used to select certain parameters necessary to operate your system in the Atari Mode. Switch SW-1-7 must be OFF (Down) when used in the Atari Mode. When Switch SW-1-7 is OFF as noted above, Switches SW-1 and SW-2 should be configured for your particular installation if you desire to operate in any mode besides the Atari Standard Mode (Single Density, No Line Printer, 5 1/4" disk drive, 1 drive available, and Base Drive Number = 1).

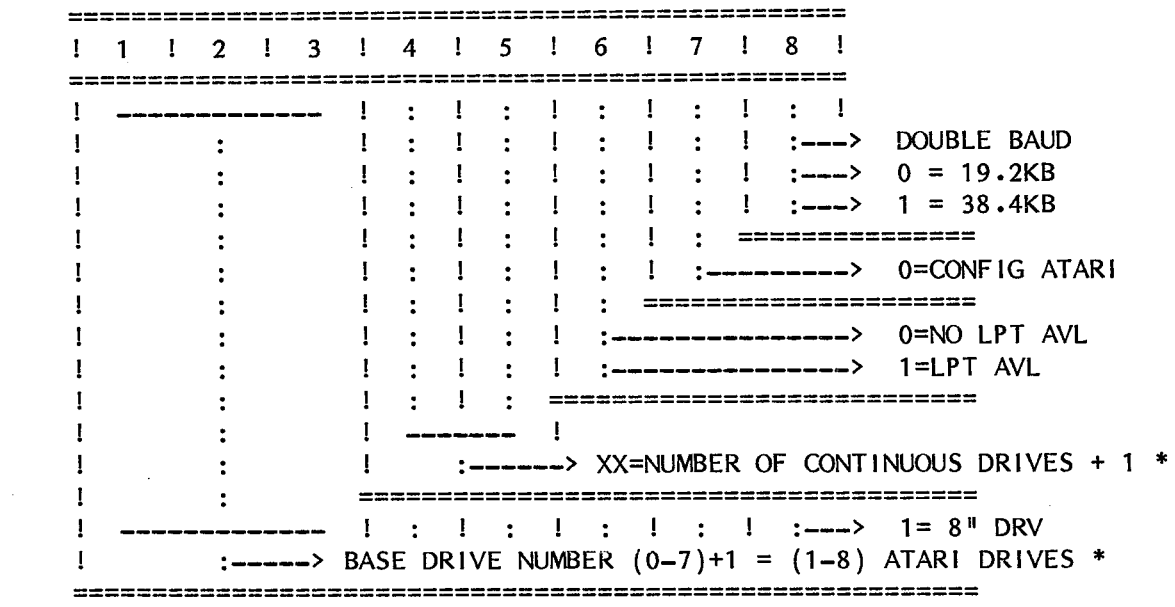
Please note that SW-1-6 enables the printer when ON (UP). SW-1 Switches 1-5 must be configured properly according to your particular installation.

Switch SW1-8 when ON (UP) allows you to configure your MF-XX8X drive for 38.4 KB (double the standard Atari rate) when you are sending information from the drive to the Atari (this is valid only for future operating systems as none of the currently available systems are capable of supporting this function). The normal position of SW1-8 will be in the OFF (down) position. The drive is shipped from the factory with the following Default settings:

SW-1 All Switches will be initially set to OFF (0).
SW-2 All Switches will be initially set to OFF (0).

These factory settings will allow your unit to emulate an Atari disk drive. To use the printer feature, double density, or 8" disk drives, you should configure the switches accordingly.

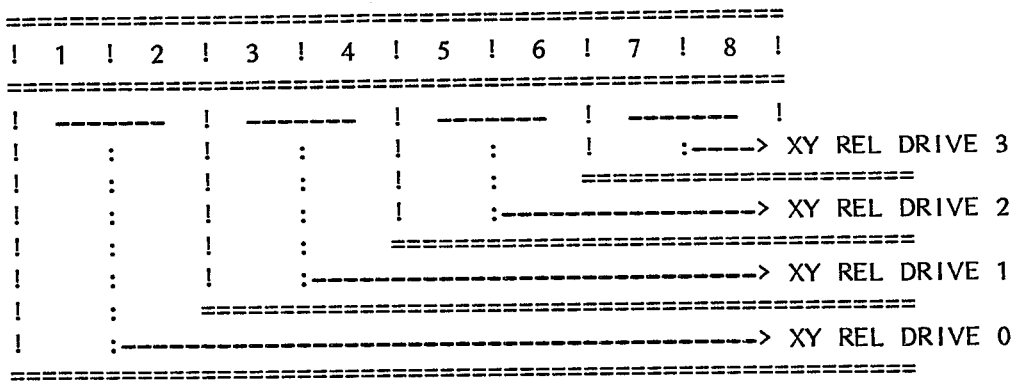
SWITCH SW-1 CONFIGURATION


1 2 3 4 5 6 7 8
DOUBLE BAUD
0 = 19.2KB
1 = 38.4KB
0=CONFIG ATARI
0=NO LPT AVL
1=LPT AVL
XX=NUMBER OF CONTINUOUS DRIVES + 1 *
1= 8" DRV
BASE DRIVE NUMBER (0-7)+1 = (1-8) ATARI DRIVES *

Where AVL = Available
DRV = Drive
* = Binary Coded

 <c> 1982, Micro Mainframe

SWITCH SW-2 CONFIGURATION



Where: X = Density of Drive (0=Single; 1=Double)
Y = Size of Drive (0=5 1/4"; 1=8")
REL = Relative

Please refer to XY Definition Below for map of definition.

Atari Drive Number = Base Drive + 1
Config for Drive = Relative Drive Switches

XY DRIVE CONFIGURATION BIT MAP

XY DEFINITION

00 5 1/4" Single Density
10 5 1/4" Double Density
01 8" Single Density
11 8" Double Density

 <c> 1982, Micro Mainframe

SECTION 3

UTILITY COMMANDS SUPPORTED

This portion of the technical manual describes some of the advanced features of the MF-168X and the MF-XX8X series disk drives for the Atari computer system by Micro Mainframe. Sample configuration procedures are given throughout. These advanced features will work only on Micro Mainframe disk drives for the Atari and are not available for any other disk drive.

You may have realized by now that the Micro Mainframe Atari disk drive is more than just a simple disk drive. It is, in reality, a powerful 16K, 4 MHz. Z80A computer system as shipped from the factory. A 64K memory expansion kit is available as well as a CPM package which will allow your Atari to operate as an inexpensive terminal for a more powerful computer system. All of this is possible without losing the basic function of performing as an Atari disk drive. Now you can have Atari compatibility when you desire, and switch to the CPM system when you want to take your computer beyond the limitations of the Atari system.

A. SOFTWARE COMMANDS SUPPORTED VIA THE LINE PRINTER CHANNEL

This series of commands will allow the user to communicate to the PCP-80 portion (computer) and the FPC-80 portion (disk controller/printer port) of the disk drive. Several useful utilities are provided for your convenience. Some of these presume a knowledge of Z-80 programming and/or technical knowledge of disk operation. ***** WARNING **** If you do not have the required knowledge, we recommend that you become familiar with these procedures before using these commands. All consequences of improper use of any of these features become the liability of the user and Micro Mainframe will not be liable for any damage to any program or diskette caused through the mis-use of any of these commands.** Provision of these commands gives the knowledgable systems programmer a degree of control over his software never before available. One example of the use of these commands might be to create custom formatted diskettes for software distribution.

B. COMMAND FORMAT

The command format for transmitting a command to the PCP-80 portion of the Micro Mainframe Disk Drive is as follows:

27, 27, Op-Code, Auxiliary Data, Carriage Return

The PCP-80 computer will intercept any line printer data which has a leading sequence of ESC ESC (27 Decimal or 1B Hex), followed by a legal op-code and auxiliary data. If only one ESC is found or the op-code is not supported, the PCP-80 will treat the data as if it is line printer data. If the PCP-80 was powered up with the line printer configuration switch (SW1-6) in the off or unavailable setting, the PCP-80 will not respond to any line printer channel commands. To issue commands when the above is true, use the disk command programming method.

When two ESC characters are detected, the op-code is examined. If the op-code is valid, then the appropriate routine is entered. If the op-code is illegal, the data is treated as line printer data.

The PCP-80 has 3 predefined device handlers. These handlers are numbered 0-2 and their functions are as follows:

HANDLER NUMBER	DESCRIPTION
00	Floppy Disk Device Handler
01	Line Printer Handler
02	Hard Disk Handler

 <c> 1983, Micro Mainframe

C. OP-CODES SUPPORTED

The following OP-CODES are currently supported by the PCP-80.

OP-CODE	DESCRIPTION
00	Define a Device.
01	Remove a Device.
02	See Note.
03	Dump PCP-80 RAM to Line Printer.
04	Modify PCP-80 RAM contents.
05	Read a track from the floppy disk.
06	Burn a track on the floppy disk.
07	Issue commands and data directly to the hard disk controller.
08	Software reset for the PCP-80 CPU.

D. OP-CODE DEFINITIONS

The following is an explanation of the individual op-codes supported.

OP-CODE 00

Each of the devices supported by the PCP-80 has different device blocks. Any device supported by the PCP-80 will have a block of data exactly 12 bytes long which contains information as to which device ID, handler, parameters for the device, time-out information for the device, and the status information needed by the Atari computer. Since this is probably somewhat confusing, let us define a floppy disk drive.

We must send 11 bytes of information to the PCP-80 when defining a device. All device description parameters will be described as relative bytes in this block.

The following example is a des Block fo Floppy Drive

RELATIVE BYTE	DESCRIPTION
00	Atari Device ID, Drives (Hex 30-37)
01	Handler Number (0-N) see above. Normally (00)
02	Drive Floppy drive type. See drive type below.
03	Number of cylinders for this drive
04	Number of sectors per track for this drive
05	Number of heads for this drive (0-1 for floppy)
06	Atari Command Status (see below)
07	Hardware status (used by Atari)
08	Device time-out value (used by Atari)
09	CRC byte of status used by Atari

Relative byte 02 has a bit mapped definition as defined below:

BIT	DEFINITION
7	Density of Drive (0=Single; 1=Double)
6	Not used
5	Not used
4	Size of Disk (0=5 1/4"; 1=8")
3	Internal
2	Not used
1-0	Step Rate for Drive

 <c> 1983, Micro Mainframe

BITS 0–1	5 1/4"	8"
00	6 MS	3 MS
01	12 MS	6 MS
02	20 MS	10 MS
03	30 MS	15 MS

Relative byte 06 must have bit 5 set for double density and reset for single density. Other bits will be used internally by the PCP–80.

NOTE 1: The number of heads should be the physical number of heads on the drive minus one. For a double headed drive, the number of heads should be set to 1. The number of sectors per track (Relative byte 04) should equal the number of sectors per side per track. The PCP–80 will automatically double the number of sectors per track when two sided operation is specified.

NOTE 2 (see NOTE 1 also): The PCP–80 is setup to use 18 sectors per track for 5 1/4" operation and 26 sectors per track for 8" operation. The PCP–80 will double the number of sectors per track for double headed operation as mentioned above.

The PCP–80 supports any number of tracks per side up to a maximum of 255 tracks, thus allowing for future high capacity floppy disk drives.

E. PROGRAMMING EXAMPLES

The following program is a sample Basic program that allows you to send a command line to your PCP–80 to configure drive number 2. NOTE: This will only work on drives that are controlled directly by the PCP–80 and will not work on your Atari or Percom drive.

EXAMPLE 1 – SINGLE DENSITY, SINGLE SIDED, 40 TRACKS, 18 SECTORS PER TRACK, AND 6 MS STEP RATE.

```
10  DATA 27, 27, 0, 50, 0, 0, 40, 18, 0, 0, 0, 255, 0, 0
20  DIM A$ (14)
30  FOR G = 1 TO 14 : READ A : A$ (G,G) = CHR$ (A) : NEXT G
40  LPRINT A$
50  END
```

EXAMPLE 2 – DOUBLE DENSITY, DOUBLE SIDED, 40 TRACKS, 18 SECTORS PER TRACK, AND 6 MS STEP RATE.

```
10  DATA 27, 27, 0, 50, 0, 128, 40, 18, 1, 32, 0, 255, 0, 0
20  DIM A$ (14)
30  FOR G = 1 TO 14 : READ A : A$ (G,G) = CHR$ (A) : NEXT G
40  LPRINT A$
50  END
```

This concludes the discussion on configuring for different floppy disk drive types, and should be self explanatory for other types of devices if you read carefully the definitions noted above.

 <c> 1983, Micro Mainframe

F. LINE PRINTER DEVICE DESCRIPTION BLOCK

This device will automatically be included in the configuration table if the line printer available switch is set on power-on or reset, otherwise it is not included.

BYTE	DESCRIPTION
00	ID. Normally (01)
01	Handler Number Normally (01)
02	Not Used
03	Not Used
04	Not Used
05	Not Used
06	Command Status
07	Last line length byte (Atari definition)
08	Time-out
09	Not Used
10	CRC byte of Status Request

G. HARD DISK CONTROL BLOCK

Although the hard disk control block is very complex, its structure allows it to be very flexible. We will not attempt to give every configuration possible. Instead, we refer you to the MMF-Atari Hard Disk Manual shipped with your MMF hard disk system.

HARD DISK DEVICE BLOCK

BYTE	DESCRIPTION
00	Device ID
01	Handler Number (02)
02	Base LUN and Step Rate (see below)
03	Low Byte of Sector Offset to start of disk area
04	High Byte of Sector Offset to start of disk area
05	Hard Disk, Controller Type, MSB of sector offset to start of area (see below)
06	Command Status (bit 5 should always be set "815")
07	Hardware Status
08	Time Out
09	Not/Used
10	CRC bytes of Status

 <c> 1983, Micro Mainframe

BIT MAP OF RELATIVE BYTE 02 - BASE LUN AND STEP RATE

BIT	DESCRIPTION
7-6	Ignored by DTC controller Step Rate for Xebec controller. Defined as follows: 00 = Standard 3 MS step rate 01 = Half Step Option (Seagate, Texas Inst) 10 = Half Step Option (Tandon) 11 = Buffered step mode (CMI, RMS, Seagate 412: 200 MS) These step options will be translated by the PCP-80 to the following and placed into the control field bits (2-0): 00 = 000 01 = 001 10 = 010 11 = 100
5-0	Value used to determine base LUN number ID-(Base LUN) = physical hard drive number (0-7)

BIT MAP OF RELATIVE BYTE 05 HARD DISK, CONTROLLER TYPE, MSB OF OFFSET

BIT	DESCRIPTION
7	Always set to indicate a hard disk.
6-5	Controller Type. Defined are: 00 = XEBEC 01 = DTC 10 = Not Defined 11 = Not Defined
4-0	MSB of Base Offset to start of area

The Offset to the start of the disk area is defined as **The Number of sectors offset from 0 to the start of the disk area for this device.** A total of 21 bits has been allocated to the offset sector number. This number is then added to the sector number desired to achieve the sector number on the disk that we wish to address. This allows either for the hard disk to be partitioned in various ways or no partitioning at all. The maximum size of a single file is therefore limited to a maximum of 65536 sectors per device partition excluding the directory or other system information. This is not a limitation of the Micro Mainframe unit, but is due to the two-byte sector addressing method used by Atari. The 21 bits allocated for sector offset gives the Micro Mainframe unit a maximum capacity of 2,097,150 sectors which may vary in length depending upon the hardware used.

H. OP-CODE DESCRIPTIONS

We will now briefly discuss each OP-CODE and give an example in Basic that should illustrate the use of each op-code.

 <c> 1983, Micro Mainframe

OP-CODE 01 - REMOVE A DEVICE

FORMAT:

27, 27, 01, (~~DEVICE NUMBER~~)

This command will remove the device number specified if the device is defined or ignore the command if the device is not currently defined. If the device exists, it is consequently removed from the Device Definition table is reduced in size by 12 bytes.

10 LPRINT CHR$(27); CHR$(27); CHR$(1); CHR$(4)

This command will remove ~~device~~ number 4 when RUN is typed.

OP-CODE 02 - NOT SUPPORTED IN THIS MANUAL

OP-CODE 03 - DUMP PCP-80 RAM

FORMAT:

27, 27, 3, ADD LOW, ADD HI, LEN LOW, LEN HI

This command will cause the PCP-80 computer to DUMP the contents of its RAM to the line printer. The auxiliary data is the RAM Address Low Byte, RAM Address High Byte, Length Low Byte, and Length High Byte. Try this example:

10 LPRINT CHR$ (27); CHR$ (27); CHR$(3); CHR$ (0); CHR$(64); CHR$(0); CHR$(1)

This example will send a Hex dump of the PCP-80 RAM to the line printer. Ram will be dumped from address 4000 Hex to ~~42FF~~ 40FF Hex, and the number of pages (256 Bytes each) will be equal to 1 (see length field).

OP CODE 04 - MODIFY RAM

FORMAT:

27, 27, 4, ADD LOW, ADD HIGH, 0 < LEN < = 32, DATA

This example will place the data bytes into the PCP-80 RAM beginning at the address specified for the length specified. Use this command with caution, as you might cause the PCP-80 to malfunction. Any use of this function is at your own risk, but we have included it for those who have the knowledge to make it function properly (ie. - knowledge of Z-80 programming and how disk systems operate internally).

10 LPRINT CHR$(27); CHR$(27); CHR$(4); ~~CHR$(0); CHR$(64); CHR$(3); "3E0D26"~~

This example will place the hex values 3E, 0D, and 26 into the PCP-80 RAM beginning at address 4000H

OP-CODE 05 - READ TRACK

FORMAT

27, 27, 5, ~~DRIVE #~~, TRACK #

Issuing this OP-CODE will cause the PCP-80 to read a track from the floppy disk drive (without regard to the type of data present) and send it to the line printer as a Hex dump. This function will allow you to determine how any track is organized on the diskette, and will print all of the information regarding sector format.

10 LPRINT CHR$(27); CHR$(27); CHR$(5); CHR$(1); CHR$(1)

This example will cause Track 1 of Drive D1 to print out in its entirety on the line printer (the line printer must be attached and enabled as per paragraph 3.0 of the operations manual before this command will work).

 <c> 1983, Micro Mainframe

OP-CODE 06 - BURN TRACK

FORMAT

27, 27, 6, DRIVE #, TRACK #

This OP-CODE will cause the PCP-80 to destroy a specified track on a diskette. The track that you wish to destroy must have been previously formatted before you issue the command. **Since this command makes the track unreadable, we advise that you use extreme caution when using it as it could render your diskette useless.** You may only use this command with the floppy disk, as it is inoperative when used in conjunction with the hard disk. The example for this command is identical to that for OP CODE 4 as noted above.

OP-CODE 07 - ISSUE COMMANDS TO THE HARD DISK CONTROLLER

FORMAT:

27, 27, 7, DIR, 0 > LEN < = 30, CMDBLK, DATA

This OP-CODE provides you with the capability of issuing commands directly to the hard disk controller.

The firmware for the hard disk has been designed for maximum flexibility. As such, it provides a step-rate field which is used by Xebec controllers for several step rates supported by that controller. The Base LUN # is used in conjunction with the device ID in order to allow multiple hard disks to be attached and selected automatically. The sector offset field is to be used for device partitioning. The hard disk type flag and controller type bits are provided to allow the user to configure for almost any SASI-BUS hard disk system. The Base LUN # field byte 3 bits 5-0 is subtracted from the device ID to achieve the physical LUN # to be addressed (where LUN = Logical Unit Number). For example, a Device ID # of 50 Hex and a Base Unit # of 49 Hex equals a LUN # of 1 (50 - 49 = 1).

Sector Offset Field bytes 4, 5, (6 <Bits 0 - 4>) represents the number of sectors offset from the start of the disk to begin this new area on the disk. The number thus obtained is added to the desired sector number to obtain the physical sector number.

One such example would be to configure the PCP-80 to handle the Xebec controller for a 10 or 15 megabyte drive. This command could also allow you to configure for other SASI-BUS * compatible controllers or different drive sizes. An explanation of the format follows:

DIR = Direction of the activity; 0=READ; 1=WRITE.
LEN = Number of data bytes supplied (<=30 Supplied). The routine will supply more if needed but the user supplied byte count is limited to 30.
CMDBLK = Hard disk controller's command block (consult the appropriate manual for the controller used).
DATA = Any data required for the operation requesting.

Please consult the manual appropriate for your hard disk for examples of the programming methods.

 <c> 1983, Micro Mainframe

OP-CODE 08 - RESET

FORMAT

27, 27, 8

This command will cause the PCP-80 to position the Firmware to the start of the code (effectively, a power on reset). If the controller is locked up because of a hardware condition, this command will have no effect and you will have to turn the power on the disk drive off and then back on to achieve a reset condition.

I. LOGICAL ADDRESS COMPUTATION

The following information is provided for those knowledgable in the methods for performing disk I/O on both floppy disk drives and hard disk drives. Not everyone will understand the purpose of this information, but it is provided for those who have a need to know.

LOGICAL ADDRESS COMPUTATION - FLOPPY DISK

ATARI SECTOR # = (CYADD + HEAD) * SECTRK + SECTOR

Where:

CYADD = CYLINDER # (0-N) DESIRED
SECTOR = SECTOR NUMBER DESIRED
HEAD = NUMBER OF HEADS PER CYLINDER (0-1)
SECTRK = NUMBER OF SECTORS PER TRACK (ONE SIDE)

Then: Resolved 4S

PHYSICAL TRACK # = INT(ATARI SECTOR / (SECTRK * (# HEADS + 1)))
SECTOR NUMBER = ATARI SECTOR - (TRACK # * (SECTRK * (# HEADS + 1)))
SIDE = IF SECTOR # > SECTRK THEN (SIDE B)

LOGICAL ADDRESS COMPUTATION - HARD DISK

LOGICAL ADR = (CYADR * HDCYL + HDADR) * SECTRK + SECADR

Where:

CYADR = CYLINDER ADDRESS
HDADR = HEAD ADDRESS
SEADR = SECTOR ADDRESS
HDCYL = NUMBER OF HEADS PER CYLINDER
SECTRK = NUMBER OF SECTORS PER TRACK

 <c> 1983, Micro Mainframe

APPENDIX B

APPLICATION NOTE: BAUD RATE SELECT

Micro MainfRAMe has incorporated within the firmware package of the MF-XX8X, a softswitch RAM location to signal to the MF-XX8X the write serial bit baud rate. The RAM location for this softswitch is at 404DH. The switch may be changed at any time and will effect only the write baud rate, NOT the receive baud rate as the AT-XX8X will track any reasonable baud rate up to 76.8k baud and possibly beyond.

There are two ways of changing the write baud rate. The first is to select either 19.2k or 38.4k by the dip switch SW1-8. When the switch is DOWN or OPEN the baud rate is 19.2 KB, the standard Atari baud. When the switch is UP or CLOSED the baud rate is 38.4 KB. Please note that switch SW1-8 is read only during initial power on or a manual reset condition which is supplied by pressing switch SW-3 on the back panel of the MF-XX8X disk drive. The value thus obtained is stored in the softswitch RAM location 404DH.

The second method of changing the baud rate is through the MOD RAM command through the line printer channel using OP-CODE 04 (please refer to Section III of the Technical Manual). The following BASIC statments demonstrate how to change the baud rate using this method. Please note that this feature will not be valid until it is supported by some vendor or vendors of operating systems. Inclusion of this feature is just one example of Micro MainfRAMe's continuing dedication to provide you with advanced features unobtainable through any other manufacturer.

TO SET 38.4 K BAUD:

```
10 LPRINT CHR$(27);CHR$(27);CHR$(4);CHR$(77);CHR$(64); CHR$(1);CHR$(255)
```

TO SET 19.2 K BAUD:

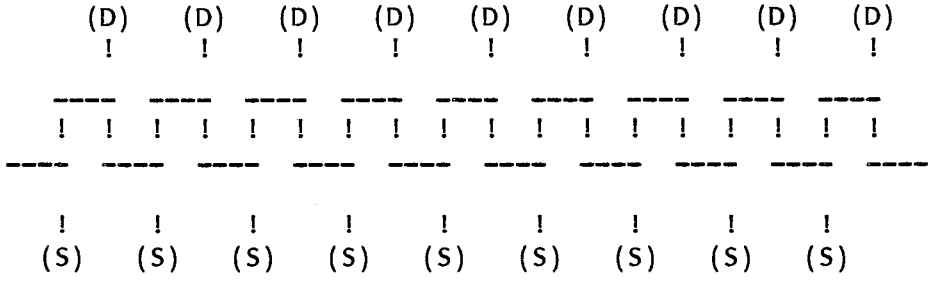
```
10 LPRINT CHR$(27);CHR$(27);CHR$(4);CHR$(77);CHR$(64); CHR$(1);CHR$(0)
```

Please remember that the read clock will track the Baud rate sent to the Atari regardless of what the write clock baud rate is set to.

We shall now briefly describe the MF-XX8X clock.

The MF-XX8X uses a true square wave clock with a 50% duty cycle. This clock is used to control the transmitting of serial data bits between the Atari and the MF-XX8X. The clock may be observed on an oscilloscope through port (0FFH) described in Section II of the manual and resembles the following drawing:

19.2 K BAUD CLOCK WAVEFORM



Each rising edge of the clock marks the (S)tart of a data bit window. Each falling edge represents the the location of the data bit or when the data (D) bit is valid.

 <c> 1983, Micro MainfRAMe